Logica Matematica 1

Davide Peccioli

Anno accademico 2022-2023

Università degli studi di Torino

Indice

Capitolo 1

Introduzione

1.1 Crisi delle fondamenta

(1.1) Alla fine del '800 nascono dei problemi con le fondamenti delle matematiche: proprio da questo nasce la Logica Matematica.

Esempio - Paradosso di Russel. (1.2) Supponiamo di avere una proprietà P. Sembra naturale che, data P, posso considerare la collezione degli insiemi che hanno tale proprietà:

$$\{x : x \text{ soddisfa } P\}$$

Posso prendere P: x non appartiene a se stesso. La collezione diventa:

$$A \coloneqq \{x : x \notin x\}$$

Domanda: è vero che $A \in A$?

- Se ipotizzo che $A \in A$

  $\Rightarrow A \notin A$: assurdo.

- Se ipotizzo che $A \notin A$

  $\Rightarrow A \in A$: assurdo.

Ottengo quindi che $A \in A \Leftrightarrow A \notin A$.

Domande e problemi. (1.3)

1. Data una dimostrazioni, posso “verificare” che sia corretta?
2. Si può dimostrare ogni affermazione vera?
3. Cos’è una dimostrazione?

1.2 Nozioni di base

Cos’è una dimostrazione. (1.4) È una catena di passaggi che parte dall’ipotesi e arriva alla tesi.

A livello teorico dovremmo poter controllare tutti i passaggi, e stabilire se sono validi o meno. Non è chiaro però cosa siano esattamente i passaggi logici.

Vale che ogni dimostrazione matematica si basa su un numero finito (poco più di una decina) di passaggi.

Logica. (1.5) La logica è un linguaggio formale, composto da

- sintassi, ovvero le “regole grammaticali”; bisogna quindi introdurre
  - simboli e lettere da utilizzare;
  - regole per la formazione di “frasi”
  - regole corrette per i “passaggi logici”
- semantica, ovvero il significato di ciò che si scrive: comprende
  - il significato dei simboli
  - l’interpretazione delle “frasi” in un dato contesto
  - la “frase” è vera o falsa?

Il linguaggio formale che studienremo si chiama logica del prim’ordine.

1.3 Logica Proposizionale

Sintassi della logica proposizionale. (1.6)

- Simboli:
$$L = \{A, B, C, \dots\}, \wedge, \vee, \neg, \Rightarrow, \Leftrightarrow, (\ ,\ )$$
dove L è l'insieme delle lettere proposizionali (ovvero il corrispettivo delle proposizioni semplici).

- Formule (proposizioni):
  1. se $A \in L$, allora (A) è una proposizione;
  2. se P è una proposizione formata, anche $(\neg P)$ lo è;
  3. se P e Q sono proposizioni, allora lo sono anche
  $$(P \wedge Q),\ (P \vee Q),\ (P \Rightarrow Q),\ (P \Leftrightarrow Q)$$

- Regole di derivazione: sono nella forma:
$$\frac{P_1 \quad P_2 \quad \dots \quad P_n}{Q}$$
Queste regole sono:
  - $\dfrac{P \wedge Q}{P}$
  - $\dfrac{P \wedge Q}{Q \wedge P}$
  - $\dfrac{P \quad Q}{P \wedge Q}$
  - $\dots$

- Dimostrazione (derivazione): applicazione delle regole di derivazione che partendo da $P_1, \dots, P_n$ mi porta a concludere Q

Semantica della logica proposizionale. (1.7) Questi sono i significati attesi per i simboli:

- $\wedge$: sta per la congiunzione,
- $\vee$: sta per la disgiunzione,
- $\neg$: sta per la negazione,
- $\Rightarrow$: sta per implicazione,

- $\Leftrightarrow$: sta per la bi-implicazione.

Questo però viene da:

- Valutazione (modello): è una funzione

  $$v : L \longrightarrow \{\boldsymbol{V}, \boldsymbol{F}\}$$

  che non è determinata dalla logica.

  Questa funzione, però, si estende a

  $$v : \{\text{Proposizioni}\} \longrightarrow \{\boldsymbol{V}, \boldsymbol{F}\}$$

  secondo le seguenti regole:

  - $v(\neg P) = \boldsymbol{V}$ esattamente quando $v(P) = \boldsymbol{F}$
  - $v\,(P \wedge Q) = \boldsymbol{V}$ esattamente quando $v(P) = v(Q) = \boldsymbol{V}$;
  - $v\,(P \vee Q) = \boldsymbol{V}$ esattamente quando $v(P) = \boldsymbol{V}$ oppure† $v(Q) = \boldsymbol{V}$;
  - $v(P \Rightarrow Q) = \boldsymbol{V}$ esattamente quando se $v(P) = \boldsymbol{V}$ allora $v(Q) = \boldsymbol{V}$;
  - $v(P \Leftrightarrow Q) = \boldsymbol{V}$ esattamente quando $v(P) = v(Q)$

- Conseguenza Logica: $P_1, \ldots, P_n \models Q$ vuol dire che se $P_1, \ldots, P_n$ sono vere, allora è vera anche Q. Formalizzando: per ogni $v : L \to \{V, F\}$, se

  $$v(P_1) = \cdots = v(P_n) = \boldsymbol{V}$$

  allora $v(Q) = \boldsymbol{V}$

Teorema I.

Teorema di Completezza per la Logica Proposizionale

$P_1, \ldots, P_n \models Q$ se e solo se esiste una derivazione di Q da $P_1, \ldots, P_n$.

† C'è una ambiguità nell'utilizzo del termine "oppure", che risolveremo in seguito.

Capitolo 2

Sintassi

Mappa. (2.1)

- Simboli.
- Termini e formule.
- Modello/struttura.
- Interpretazione di fml nelle strutture.
- "Verità".
- Derivazioni.

2.1 Simboli

Alcuni simboli già utilizzati. (2.2)

- Variabili: $x, y, z, \dots;$
- costanti: $\pi, e, i, \dots, 0, 1, \dots;$
- funzioni: $f, g, h, \dots, +, \cdot;$
- relazione: $\leq, <, >, \dots;$

Soffermiamoci inoltre su:

- connettivi: e, o, non, se…allora, ...se e solo se...

- quantificatori: per ogni $x\dots$, esiste $x\dots$

La differenza sostanziale tra i due gruppi di simboli individuati è che le componenti del primo variano da contesto a contesto, mentre quelli del secondo gruppo sono trasversali, e si utilizzano in ogni contesto.

Formalizzeremo definendo gli elementi del primo gruppo come linguaggio, mentre quelli del secondo gruppo come costanti logiche[†].

Connettivi. (2.3)

1. Congiunzione ("e"): $\wedge$.

   $\phi \wedge \psi \rightsquigarrow$ vale ϕ e vale ψ

   $$\frac{\phi \wedge \psi}{\phi} \qquad \frac{\phi \wedge \psi}{\psi} \qquad \frac{\psi \quad \phi}{\phi \wedge \psi} \qquad \Bigg| \qquad \frac{\phi \wedge \psi}{\psi \wedge \phi}$$

   L'ultima affermazione, però, è derivabile dalle altre:

   $$\dfrac{\dfrac{\phi \wedge \psi}{\phi} \qquad \dfrac{\phi \wedge \psi}{\psi}}{\psi \wedge \phi}$$

2. Disgiunzione ("o" inclusivo): $\vee$

   $$\begin{aligned} \phi \wedge \psi &\rightsquigarrow \text{vale almeno una tra } \psi \text{ e } \phi \\ &\rightsquigarrow \text{vale } \psi \text{ oppure vale } \phi\text{, oppure entrambe.} \end{aligned}$$

   Le principali regole sono

   $$\frac{\psi}{\phi \vee \phi} \qquad \frac{\phi}{\phi \vee \phi} \qquad \frac{\phi \vee \psi \qquad \text{non } \phi}{\psi}$$

3. Negazione ("non"): $\neg$

   $\neg\,\phi \rightsquigarrow \phi$ non è vera.

   Le principali regole sono

   $$\frac{\phi}{\neg\neg\,\phi} \qquad \frac{\neg\neg\,\phi}{\phi}$$

[†] "costanti" logiche in quanto hanno lo stesso significato in tutte le branche della matematica.

4. Implicazione ("se. . . allora. . . "): $\Rightarrow$

   $\phi \Rightarrow \psi \rightsquigarrow$ se vale ϕ allora vale ψ.

   Notiamo che l'affermazione $\phi \Rightarrow \psi$ è falsa soltanto se ϕ è vera mentre ψ è falsa. Dunque si ha:

   $\phi \Rightarrow \psi \rightsquigarrow$ o non vale ϕ oppure vale ψ.

   Le principali regole sono:

$$\frac{\phi \Rightarrow \psi}{\neg\phi \vee \psi} \qquad \frac{\neg\phi \vee \psi}{\phi \Rightarrow \psi}$$

5. Bi-implicazione (". . . se e solo se. . . ") $\Leftrightarrow$

   $\phi \Leftrightarrow \psi \rightsquigarrow$ o ϕ e ψ sono vere o ϕ e ψ sono false.

   Le principali regole sono:

$$\frac{\phi \Leftrightarrow \psi}{(\phi \Rightarrow \psi) \wedge (\psi \Rightarrow \phi)} \qquad \frac{\phi \Rightarrow \psi \qquad \psi \Rightarrow \phi}{\phi \Leftrightarrow \psi}$$

Esempio - Leggi di De Morgan. (2.4) Le Leggi di De Morgan affermano che

$$\frac{\phi \wedge \psi}{\neg(\neg\phi \vee \neg\psi)} \qquad \frac{\neg(\neg\phi \vee \neg\psi)}{\phi \wedge \psi} \tag{2.1}$$

$$\frac{\phi \vee \psi}{\neg(\neg\phi \wedge \neg\psi)} \qquad \frac{\neg(\neg\phi \wedge \neg\psi)}{\phi \vee \psi} \tag{2.2}$$

e si possono derivare dalle regole viste finora

Esempio - Altri connettivi. (2.5) Ecco una lista di altri connettivi che non useremo:

- XOR $\oplus$: $\psi \oplus \phi \rightsquigarrow \phi$ o ψ ma non entrambi.

  $\psi \oplus \phi \leftrightsquigarrow (\psi \vee \phi) \wedge \neg(\psi \wedge \phi)$.

- NOR: ψ NOR $\phi \rightsquigarrow$ né ψ né ϕ.

  ψ NOR $\phi \leftrightsquigarrow (\neg\psi) \wedge (\neg\phi)$

Quantificatori. (2.6)

1. Quantificatore universale ("per ogni…"): $\forall$

   $\forall x\, \psi \rightsquigarrow$ per tutti gli x vale ψ.

2. Quantificatore esistenziale ("esiste x…"): $\exists$

   $\exists x\, \psi \rightsquigarrow$ esiste almeno un x tale che ψ

Le principali regole sono

$$\frac{\neg\,(\forall x\, \psi)}{\exists x\,(\neg\,\psi)} \qquad \frac{\exists x\,(\neg\,\psi)}{\neg\,(\forall x\, \psi)} \qquad \frac{\neg\,(\exists x\, \psi)}{\forall x\,(\neg\,\psi)} \qquad \frac{\forall x\,(\neg\,\psi)}{\neg\,(\exists x\, \psi)}$$

Inoltre vale:

$$\frac{\exists x\,\exists y\,\psi}{\exists y\,\exists x\,\psi} \qquad \frac{\forall x\,\forall y\,\psi}{\forall y\,\forall x\,\psi} \qquad \frac{\exists x\,\forall y\,\psi}{\forall y\,\exists x\,\psi} \qquad \cancel{\frac{\forall y\,\exists x\,\psi}{\exists x\,\forall y\,\psi}}$$

In relazione alla congiunzione e la disgiunzione, vale:

$$\frac{\exists x\,\psi \qquad \exists x\,\phi}{\exists x\,(\psi \vee \phi)} \qquad \frac{(\exists x\,\psi) \vee (\exists x\,\phi)}{\exists x\,(\psi \vee \phi)} \qquad \frac{\exists x\,(\psi \vee \phi)}{(\exists x\,\psi) \vee (\exists x\,\phi)}$$

$$\frac{\forall x\,\psi \qquad \forall x\,\phi}{\forall x\,(\psi \wedge \phi)} \qquad \frac{(\forall x\,\psi) \wedge (\forall x\,\phi)}{\forall x\,(\psi \wedge \phi)} \qquad \frac{\forall x\,(\psi \wedge \phi)}{(\forall x\,\psi) \wedge (\forall x\,\phi)}$$

$$\cancel{\frac{(\exists x\,\psi) \wedge (\exists x\,\phi)}{\exists x\,(\psi \wedge \phi)}} \qquad \frac{\exists x\,(\psi \wedge \phi)}{(\exists x\,\psi) \wedge (\exists x\,\phi)}$$

$$\frac{(\forall x\,\psi) \vee (\forall x\,\phi)}{\forall x\,(\psi \vee \phi)} \qquad \cancel{\frac{\forall x\,(\psi \vee \phi)}{(\forall x\,\psi) \vee (\forall x\,\phi)}}$$

Linguaggio. (2.7) Il linguaggio $\mathcal{L}$ è una lista di simboli fissata, composta da una parte fissa, ed una "mobile".

La parte fissa è composta da simboli che sono presenti in qualisiasi logica del prim'ordine:

- parentesi: (,);
- connettivi: $\wedge, \vee, \neg, \Rightarrow, \Leftrightarrow$

- quantificatori: $\forall, \exists$
- variabili: $v_0, v_1, \ldots$ [†]

La parte "mobile", invece, dipende dalla parte della matematica che si studia:

- costante: $a, b, c, \ldots$
- funzione: $f, g, h, \ldots$
- relazione: $P, Q, R, \ldots$ e $\doteq$.

In realtà $\doteq$ deve esserci sempre, ma essendo anche una relazione lo abbiamo inserito nella parte "mobile".

Chiamiamo Const l'insieme delle costanti, Fun l'insieme delle funzioni e Rel l'insieme delle relazioni.

Per semplicità possiamo considerare

$$\mathcal{L} = \mathrm{Const} \cup \mathrm{Fun} \cup \mathrm{Rel}$$

Arietà. (2.8) Sia delle funzioni che delle relazioni, è necessario definirne la arietà, ovvero il numero di input:

- unaria (arietà 1) $\rightsquigarrow$ un solo input;
- binaria (arietà 2) $\rightsquigarrow$ due input

Si scrive $\mathrm{ar}(g) = 5$

Esempio. (2.9) Consideriamo

$$\mathcal{L} = \{P, Q, g, b\}$$

con

- P simbolo di relazione binario,
- Q simbolo di relazione unario,
- g simbolo di funzione unario,
- b simbolo di costante.

[†] Si definisce in questo modo per avere a disposizione una quantità infinita di variabili.

Esempio - Linguaggio dei gruppi. (2.10) Consideriamo

$$\mathcal{L} = \{f, g, c\}$$

con

- f simbolo di funzione binario,
- g simbolo di funzione unario,
- c simbolo di costante.

2.2 Termini

$\mathcal{L}$-termini. (2.11) Sono termini:

- costanti e variabili;
- se f è un simbolo di funzione con $\mathrm{ar}(f) = n$ e $t_1, \ldots, t_n$ sono termini, allora è termine:

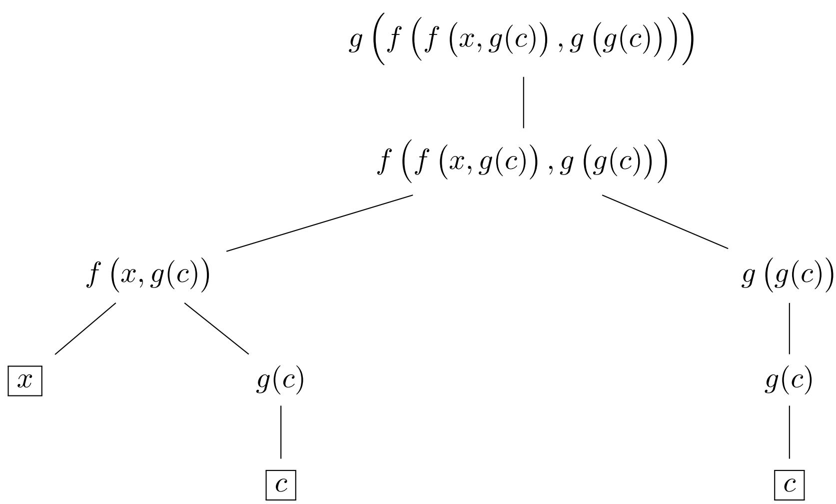
$$f(t_1, \ldots, t_n)$$

Albero sintattico. (2.12) L'albero sintattico è un diagramma che a partire dalla stringa data cerca di ricostruire le operazioni utilizzate per costruirla. Questo permette di stabilire se una stringa è un termine o meno.

Esempio. (2.13) Costruiamo l'albero sintattico, tramite il linguaggio dei gruppi, per la seguente stringa:

$$g\left(f\left(f\left(x, g(c)\right), g\left(g(c)\right)\right)\right)$$

L'albero diventa:

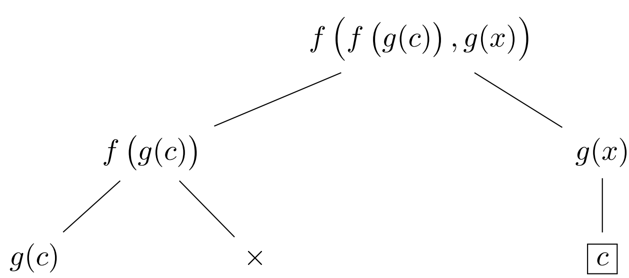


Da qui posso affermare che questo è un termine, poiché so ricostruirlo "al contrario".

Non Esempio. (2.14) Sempre considerando il linguaggio dei gruppi, mostriamo che

$$f\Big(f\left(g(c)\right), g(x)\Big)$$

non è un termine.



Definizione. (2.15) *L'altezza di un termine t è l'altezza (ovvero la lunghezza del ramo più lungo partendo da 0) del suo albero sintattico. Si scrive* $\mathrm{ht}(t)$

Definizione. (2.16) *Sia $t(x_1, \dots, x_n)$ un termine, dove $x_1, \dots, x_n$ sono le variabili che compaiono in t, e dati $s_1, \dots, s_n$ termini, la scrittura:*

$$t\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right]$$

è il termine ottenuto da t sostituendo ogni x_i con s_i.

Esempio. (2.17) Sia $f : \mathbb{R} \to \mathbb{R}$ una funzione, e $0, 1, 2, \dots$ le costanti.

Sia t il termine $f(x)$, e sia s il termine $f(1)$.

Il termine $f\left(f(1)\right)$ è ancora un termine, e lo scriviamo

$$f\left(f(1)\right) \rightsquigarrow t[s/x]$$

Lemma. (2.18) $t\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right]$ è un termine.

Dimostrazione di (2.18) Lo dimostriamo per induzione sull'altezza.

- Caso base: se $\mathrm{ht}(t) = 0$ allora ci sono due casi:
  - t è un simbolo di costante:

    $$t\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right] = t;$$

  - t è un simbolo di variabile x_i:

    $$t\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right] = s_i.$$

  Poiché t e s_i per ipotesi sono termini, ho dimostrato il caso base.

- Passo induttivo: suppongo che il lemma sia vero per tutti i termini di altezza $\leq n$ e prendo t di altezza $n + 1$.

  t è nella forma $f(u_1, \dots, u_k)$ per qualche $f \in \mathcal{L}$ simbolo di funzionio con $\mathrm{ar}(f) = k$ e $u_1, \dots, u_k$ termini di altezza $\leq n$.

  Dunque:

  $$t\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right] = f\left(u_i\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right]\right)$$

  Per l'ipotesi induttiva, gli $u_i\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right]$ sono ancora termini, e quindi anche $t\left[s_1/x_1,\, s_2/x_2, \dots,\, s_n/x_n\right]$ è termine. ■

2.3 Formule atomiche e Formule

Costruzione delle formule atomiche. (2.19)

- Dati due termini t_1 e t_1,
$$t_1 \doteq t_2$$
è una formula atomica.

- Dato $R \in \mathrm{Rel}$ di arietà n, dati $t_1, \ldots, t_n$ termini,
$$R(t_1, \ldots, t_n)$$
è una formula atomica.

Osservazione. (2.20) A differenza dei termini, le formule atomiche non sono costruite per ricorsione, ma sono solamente quelle presentate sopra.

Costruzione delle formule. (2.21)

- Se φ è una formula, lo è anche $(\neg\,\varphi)$.
- Se φ e ψ sono formule e $\odot$ è uno tra $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$, allora $(\varphi \odot \psi)$ è una formula.
- Se φ è una formula, x variabile e Q è uno tra $\forall$, $\exists$, allora $Q\,x\,\varphi$ è una formula.

Esempio. (2.22) Definisco il linguaggio:
$$\mathcal{L} = \{R, f, c\}$$
dove R è una relazione binaria, f è una funzione unaria e c è costante.

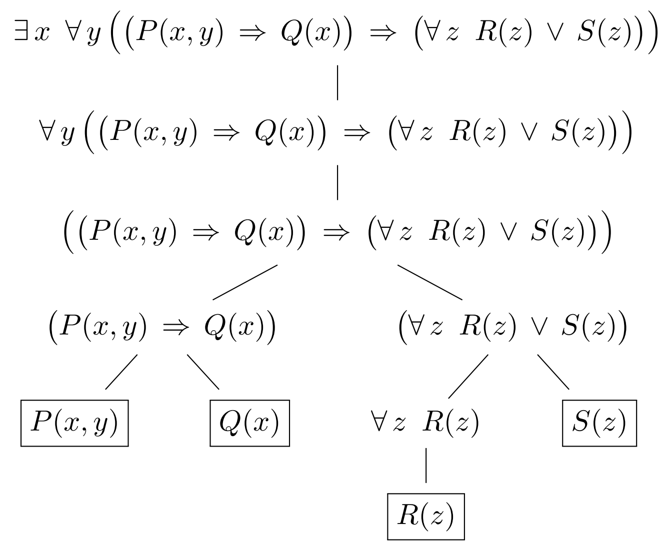
- Termini:
$$x,\ y,\ c,\ f(x),\ f(c),\ f\left(f(y)\right), \ldots$$
- Formule atomiche:
$$f(x) \doteq c,\ x \doteq f\left(f(y)\right),\ R(x,\ y),\ R\Big(f(c),\ f\left(f(y)\right)\Big).$$

- <u>Formule</u>:

$$
\begin{gathered}
\big(\neg R(x,y)\big), \quad \big(f(x) = y \wedge R(y,x)\big), \\
\Big(\big(\neg R(x,y)\big) \Rightarrow \big(f(x) = y \wedge R(y,x)\big)\Big), \\
\forall x \;\; x = f\big(f(y)\big), \quad \exists z \;\; x = f\big(f(y)\big), \\
\exists x \;\; \Big(\big(\neg R(x,y)\big) \Rightarrow \big(f(x) = y \wedge R(y,x)\big)\Big), \\
\forall y \;\exists x \;\; \Big(\big(\neg R(x,y)\big) \Rightarrow \big(f(x) = y \wedge R(y,x)\big)\Big)
\end{gathered}
$$

Osservazione. (2.23) Per verificare la correttezza di una formula si utilizza un albero sintattico

Esempio. (2.24) Dato il linguaggio $\mathcal{L} = \{P, Q, R, S\}$, tutti simboli di relazione di arietà 1, tranne P che è simbolo di relazione di arietà 2, studiamo la formula:



Altezza di una formula. (2.25) Come per i termini, si utilizza la lunghezza del ramo più lungo nell'albero sintattico di una formula (partendo a contare dallo 0), per determinare l'<u>altezza</u> di una formula.

Definizione. (2.26) *Una variabile che compare n volte in una formula si dice avere n occorrenze. Ci si riferisce alla prima occorrenza di una variabile, seconda occorrenza di una variabile, ...*

Definizione. (2.27) *Quando si applica un quantificatore ad una variabile x e ad una formula φ, φ si dice raggio di azione del quantificatore.*

Definizione. (2.28) *Quando si applica un quantificatore ad una variabile x e ad una formula φ, tutte le variabili x contenute in φ si dicono vincolate.*

Definizione. (2.29) *Quando una variabile non è vincolata si dice libera.*

Notazione. (2.30) Data una formula φ, con la scrittura

$$\varphi(x_1, \ldots, x_n)$$

si intende che se ci sono variabili libere in φ, allora sicuramente sono tra $x_1, \ldots, x_n$.

Definizione. (2.31) *Si dice enunciato una formula priva di variabili libere.*

Definizione. (2.32) *Si dice sottoformula di una formula φ del primo ordine una qualsiasi formula che è necessaria per costruire la formula φ, compresa φ.*

Osservazione. (2.33) Tutte e solo le sottoformule sono quelle che compaiono nei nodi dell'albero sintattico di una formula.

Definizione. (2.34) *Si dice sottoformula principale di una formula è l'ultima (o le ultime due) utilizzata per costruire la formula*

Definizione. (2.35) *Si dice costante logica principale l'ultima costante logica utilizzata per la costruzione della formula*

Esempio. (2.36) Data la formula $(\varphi \wedge \psi)$, si ha

$$\begin{tikzcd}[column sep=small] & (\varphi \wedge \psi) \arrow[dl, dash] \arrow[dr, dash] & \\ \varphi & & \psi \end{tikzcd}$$

e ha come costante principale $\wedge$.

Convezioni. (2.37)

1. Per le funzioni binarie, posso usare la notazione infissa.

   Es: $x + y$ al posto di $+(x, y)$.

2. Per "applicazioni" successive dello stesso simbolo di funzione binario uso l'associatività a destra.

   Es: $x + y + z = x + (y + z)$

3. Lo stesso per relazioni binarie.

   Es: $x < y < z \rightsquigarrow x < y \wedge y < z$.

4. Priorità tra le costanti logiche, con il seguente ordine:

   $$\neg,\ \forall,\ \exists$$
   $$\wedge,\ \vee$$
   $$\Rightarrow,\ \Leftrightarrow$$

   Per stessa priorità al livello 1, lega più strettamente ciò che è a destra.

2.4 Formalizzazione

Definizione. (2.38) *Per formalizzazione si intende il processo di traduzione di una qualche affermazione dal linguaggio naturale ad un linguaggio artificiale (come appunto la logica del primo ordine).*

Osservazione. (2.39) La formalizzazione non è necessariamente unica.

Osservazione. (2.40) Per formalizzare una frase è necessario avere:

- un linguaggio $\mathcal{L}$, ovvero un elenco dei simboli ammessi (sintassi);
- il significato di ciascun simbolo (semantica, ovvero $\mathcal{L}$-struttura);
- universo del discorso, ovvero l'insieme degli elementi su cui variano le variabili.

Esempio. (2.41) "Il prodotto di due numeri è zero se e solo se almeno uno dei due è zero."

Per la formalizzazione scelgo il seguente linguaggio:

$$\mathcal{L} = \{f, c\},$$

con f simbolo di funzione binario e c simbolo di costante, dove voglio interpretare f come prodotto e c come zero. La formalizzazione diventa

$$\forall x \forall y \ \big(f(x, y) = c \Leftrightarrow x = c \vee y = c\big)$$

Qual è la differenza tra questa formula e quella "senza quantificatori"?

$$\big(f(x, y) = c \Leftrightarrow x = c \vee y = c\big)$$

Questa seconda formula non è un enunciato, ovvero ha solo variabili libere: non ha senso chiedersi se questa formula è vera o falsa, ma ha solo senzo chiedersi <u>per</u> quali valori è vera o falsa.

Esempio. (2.42) "Ci sono infiniti numeri primi"

1. Scelgo come linguaggio $\mathcal{L} = \{P, <\}$ con
   - P simbolo di relazione binario,
   - $<$ simbolo di relazione binario,

   e con significato:
   - $P(x)$: "x è un numero primo",
   - "$<$" simbolo di minore.

   La formalizzazione diventa

$$\forall x \exists y \ \big(x < y \wedge P(y)\big)$$

Questa però non è una traduzione "letterale", ma ho scritto una cosa equivalente.

2. Scelgo come linguaggio $\mathcal{L} = \{|, <, 1\}$ con
   - $|$ simbolo di relazione binaria,
   - $<$ simbolo di relazione binaria,
   - 1 simbolo di costante

   e significato
   - $|$ relazione di divisibilità;
   - $<$ simbolo di minore;
   - 1 il numero uno.

   Sorge la domanda intermedia: come scrivo "y è primo"?

   $$1 < y \land \forall z \left(z \,|\, y \Rightarrow z = 1 \lor z = y \right)$$

   La formalizzazione, quindi, in questo linguaggio, sarà:

   $$\forall x \, \exists y \left(x < y \land 1 < y \land \forall z \left(z \,|\, y \Rightarrow z = 1 \lor z = y \right) \right)$$

Esempio. (2.43) "Per ogni $n > 1$ c'è almeno un primo tra n e $2n$"

- Linguaggio:
  $$\mathcal{L} = \{|, +, <, 1\}$$
  con
  - "$|$" relazione binaria
  - "$+$" funzione binaria;
  - "$<$" relazione binaria;
  - "1" costante.
- Significato: tutti i simboli hanno il loro significato naturale.
- Universo: $\mathbb{N}$.

Una prima bozza di formalizzazione è

$$\forall x \; \big(1 < x \Rightarrow \exists y \; (\text{“}y \text{ è primo”} \wedge x < y \wedge y < x + x)\big)$$

Per scrivere “y è primo” posso scrivere

$$1 < y \wedge \forall z \; \big(z \mid y \Rightarrow z \doteq 1 \vee z \doteq y\big)$$

Per comodità di notazione, posso chiamare la formula precedente $\varphi(y)$, è quindi formalizzare l'intera formula è

$$\forall x \; \Big(1 < x \Rightarrow \exists y \; \big(\varphi(y) \wedge x < y \wedge y < x + x\big)\Big)$$

2.4.1 Esempi fondamentali

Quantificatore universale limitato. (2.44) Ua frase nella forma

$\forall x$ tale che $P(x)$ vale ...

può essere formalizzata come:

$$\forall x \, (P(x) \Rightarrow \ldots)$$

Quantificatore esistenziale limitato. (2.45) Una frase nella forma

$\exists x$ per cui $P(x)$, tale che ...

può essere formalizzata come:

$$\exists x \, (P(x) \wedge \ldots)$$

Esistenza e unicità. (2.46) Una frase nella forma

Esiste un unico x tale che $P(x)$

può essere formalizzata come:

$$\exists x \; \Big(P(x) \wedge \neg \exists y \; \big(\neg (y \doteq x) \wedge P(y)\big)\Big)$$

$$\exists x \; \Big(P(x) \wedge \forall y \; \big(P(y) \Rightarrow y \doteq x\big)\Big)$$

$$\exists x \, P(x) \wedge \forall y \forall z \; \big(P(y) \wedge P(z) \Rightarrow y \doteq z\big)$$

Esistenza di almeno n oggetti. (2.47) Una frase nella forma

Esistono almeno n oggetti x tali che $P(x)$

può essere formalizzata come:

$$\exists x_1 \exists x_2 \dots \exists x_n \\ \big(P(x_1) \wedge P(x_2) \wedge \dots \wedge \neg (x_1 = x_2) \wedge \neg (x_1 = x_3) \wedge \dots\big)$$

Esistenza di al massimo n oggetti. (2.48) Una frase nella forma

Esistono al massimo n oggetti x tali che $P(x)$

può essere formalizzata come

Non esistono almeno $n+1$ oggetti x tali che $P(x)$

Esistenza di esattamente n oggetti. (2.49) Una frase nella forma

Esistono esattamente n oggetti x tali che $P(x)$

può essere formalizzata come:

Esistono almeno n oggetti x tali che $P(x)$ e esistono al massimo $n+1$ oggetti x tali che $P(x)$

Viceversa. (2.50) Quando si utilizza il "viceversa", è necessario scrivere due volte l'affermazione, esplicando quindi il significato di quel termine.

Osservazione. (2.51) Se nella frase nel linguaggio naturale compare una variabile non vincolata (esplicitamente o implicitamente) da un qualche quantificatori, allora nella sua formalizzazione tale variabile sarà libera.

Altrimenti sarà vincolata.

Esempio. (2.52) "n è un numero maggiore di 1"

$\rightsquigarrow$ $1 < n$ considerando il linguaggio $\mathcal{L} = \{<, 1\}$

"Ci sono n maggiori di 1"

$\rightsquigarrow$ $\exists n\,(1 < n)$ considerando il linguaggio $\mathcal{L} = \{<, 1\}$

2.5 Sostituzione

Definizione. (2.53) *Data una formula logica che contiene delle variabili vincolate, si chiamano varianti quelle formule ottenute dalla prima sostituendo una qualsiasi variabile vincolata con un'altra (non presente come variabile libera nella formula).*

Esempio. (2.54) Sono varianti della stessa formula:

$$\exists\, y \left((2 \cdot y) + 1 = x\right), \qquad \exists\, z \left((2 \cdot z) + 1 = x\right).$$

Sostituzione nelle formule. (2.55) Si vuole applicare lo stesso principio che si utilizza per i termini anche per le formule (utilizzando la stessa notazione). Si devono fissare alcune regole.

1. Si sostituiscono solo le occorrenza libere delle variabili con dei termini.
2. Il termine s è sostituibile per x in ϕ se nessuna delle sue variabili viene vincolata una volta effettuata la sostituzioni.

Prassi. (2.56) Data una formula φ, una variabile x e il termine s, considero una variante φ' di φ, in cui rinomino tutte le variabili vincolate in modo che siano diverse dalle variabili libere di φ, e anche dalle variabili che compaiono nel termine s.

Esempio. (2.57) Consideriamo la formula $\varphi(y)$:

$$\exists\, x\, \forall\, y \left(R(x,y) \;\Rightarrow\; R(y,x)\right) \wedge \exists\, x\, R(x,y)$$

L'unica variabile libera è la y "al fondo".

Voglio effettuare la sostituzione $\varphi\left[f(x)/y\right]$.

Scrivo una variante $\varphi'(y)$:

$$\exists\, v\, \forall\, x \left(R(v,z) \;\Rightarrow\; R(z,v)\right) \wedge \exists\, u\, R(u,y)$$

e posso ora sostituire: $\varphi'[f(x)/y]$

$$\exists\, v\, \forall\, x \left(R(v,z) \;\Rightarrow\; R(z,v)\right) \wedge \exists\, u\, R\left(u, f(x)\right)$$

Capitolo 3

Semantica

Definizione. (3.1) *Sia $\mathcal{L}$ un linguaggio del primo ordine,*

$$\mathcal{L} = \mathrm{Rel} \cup \mathrm{Fun} \cup \mathrm{Const}$$

Una $\mathcal{L}$-struttura $\mathcal{A}$ è composta da:

- *un insieme $A \neq \emptyset$, detto <u>dominio</u> o <u>universo</u>;*
- *per ogni $f \in \mathrm{Fun}$ di arietà n, una funzione*

  $$f^{\mathcal{A}} : A^n \to A$$

  $f^{\mathcal{A}}$ è chiamata <u>interpretazione</u> di f, e deve essere totale, ovvero definite su tutto A^n;
- *per ogni $R \in \mathrm{Rel}$ di arietà n, una relazione n-aria $R^{\mathcal{A}} \subseteq A^n$ su A;*
- *per ogni $c \in \mathrm{Const}$, un elemento $c^{\mathcal{A}} \in A$.*

Dunque una tipica $\mathcal{L}$-struttura con $\mathcal{L} = \{P, Q, \ldots, f, g, \ldots, c, d, \ldots\}$ sarà della forma:

$$\mathcal{A} = (A, P^{\mathcal{A}}, Q^{\mathcal{A}}, \ldots, f^{\mathcal{A}}, g^{\mathcal{A}}, \ldots, c^{\mathcal{A}}, d^{\mathcal{A}}, \ldots)$$

Esempio. (3.2) Sia $\mathcal{L} = \{f\}$, f simbolo di funzione binario. Alcuni esempi sono i seguenti.

- $\mathcal{A} = (\mathbb{N}, +)$, dove $+$ è $f^{\mathcal{A}} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$.
- $\mathcal{B} =$ (“retta”, “punto medio”) $= (\mathbb{R}, f^{\mathcal{B}})$ dove

$$\begin{aligned} f^{\mathcal{B}} : \mathbb{R}^2 &\longrightarrow \mathbb{R} \\ (x, y) &\longmapsto \frac{x+y}{2}. \end{aligned}$$

Un non esempio, invece, è il seguente:

- $\mathcal{C} = (\mathbb{N}, -)$, poiché $-$ non è definito su tutto $\mathbb{N}^2$

Esempio. (3.3) Sia $\mathcal{L} = \{R\}$ simbolo di relazione binario. Alcuni esempi sono i seguenti.

- $\mathcal{A} = (\mathbb{N}, \leq)$, dove $\leq$ è $R^{\mathcal{A}}$.
- $\mathcal{B} = (\mathbb{N}, |)$, dove $|$ è la relazione di divisibilità ed è $R^{\mathcal{A}}$.
- $\mathcal{C} = (T, R^{\mathcal{C}})$, dove T è l’insieme degli abitanti di Torino, e $R^{\mathcal{C}}$ è tale per cui $a\, R^{\mathcal{C}}\, b$ se e solo se a e b sono parenti.

3.1 Esempi di $\mathcal{L}$-strutture

Gruppi. (3.4) Un gruppo è un insieme G dotato di una operazione binaria $*$ associativa, che ha elemento neutro e inversi.

- Se $\mathcal{L} = \{*\}$, con $*$ simbolo di funzione binario.

  Un gruppo è una $\mathcal{L}$-struttura che soddisfa i seguenti $\mathcal{L}$ enunciati:

  - $\forall x \forall y \forall z \left((x * y) * z \doteq x * (y * z)\right)$;
  - $\exists z \forall x \,(x * z \doteq x \wedge z * x \doteq x)$;
  - $\exists z \left(\forall x \,(x * z \doteq x \wedge z * x \doteq x) \wedge \forall y \exists w \,(y * w \doteq z \wedge w * y \doteq z)\right)$.

- Se $\mathcal{L}_{Gp} = \{*, {}^{-1}, 1\}$, con $*$ simbolo di funzione binario, ${}^{-1}$ simbolo di funzione unario (e scriveremo x^{-1} invece di ${}^{-1}(x)$), 1 simbolo di costante.

  Un gruppo è una $\mathcal{L}_{Gp}$-struttura che soddisfa i seguenti $\mathcal{L}_{Gp}$-enunciati:

- $\forall x \forall y \forall z \big((x * y) * z \doteq x * (y * z)\big)$;
- $\forall x (x * 1 \doteq x \wedge 1 * x \doteq x)$;
- $\forall x \big(x^{-1} * x \doteq 1 \wedge x * x^{-1} \doteq 1\big)$

Se voglio parlare di gruppi abeliani, aggiungerò a questi enunciati il seguente:

$$\forall x \forall y (x * y \doteq y * x)$$

Campo. (3.5) Un campo è un insieme $\mathbb{K}$ dotato di $+$, $-$, $\cdot$, $0, 1$ tale che:

1. $(\mathbb{K}, +, -, 0)$ è gruppo abeliano;
2. $(\mathbb{K} \setminus \{0\}, \cdot, 1)$ è gruppo abeliano;
3. $\cdot$ è distributivo rispetto alla somma.

Il linguaggio tipico dei gruppi è:

$$\mathcal{L}_{Fd} = \{+, -, \cdot, 0, 1\}$$

dove $+, \cdot$ sono funzioni binarie, $-$ è funzione unaria, 0 e 1 sono costanti.

La formalizzazione sarà:

1. uguale a sopra;
2. qui è peculiare notare che nella scrittura non si è inserito il simbolo di funzione inversa di $\cdot$: infatti, tutte le funzioni devono essere totali, ovvero definite su tutto l'insieme, e l'inversa rispetto al prodotto non è mai definita su 0;
   - $\forall x \forall y \forall z \begin{pmatrix} \neg (x \doteq 0) \wedge \neg (y \doteq 0) \wedge \neg (z \doteq 0) \\ \Rightarrow (x \cdot y) \cdot z \doteq x \cdot (y \cdot z) \end{pmatrix}$
   - $\cdot$ è abeliano;
   - 1 è elemento neutro;
   - $\forall x \big(\neg (x \doteq 0) \Rightarrow \exists y (y \cdot x \doteq 1)\big)$

Campo ordinato. (3.6) Un campo ordinato è un campo $\mathbb{K}$ con una relazione di ordine lineare (totale), tale che

- se $a \leq b$ allora $a + c \leq b + c$;

- se $0 \leq a$ e $0 \leq b$, allora $0 \leq a \cdot b$.

Come linguaggio scegliamo

$$\mathcal{L} = \{+, -, \cdot, 0, 1, \leq\}$$

e i campi ordinati sono $\mathcal{L}$-strutture che sono campi, e in più soddisfano:

- ordine: $\forall x \forall y \forall z \begin{pmatrix} x \leq x \wedge (x \leq y \wedge y \leq x \Rightarrow x = y) \\ \wedge (x \leq y \wedge y \leq z \Rightarrow x \leq z) \end{pmatrix}$;
- totale: $\forall x \forall y (x \leq y \vee y \leq x)$;
- $\forall x \forall y \forall z (x \leq y \Rightarrow x + z \leq y + z)$;
- $\forall x \forall y \, (0 \leq x \wedge 0 \leq y \Rightarrow 0 \leq x \cdot y)$.

Grafo. (3.7) Un grafo è composto da un insieme V di vertici, e da una relazione binaria E di vicinanza o adiacenza (ovvero c'è un lato tra i due punti): questa relazione può valere solo tra elementi diversi (proprietà irriflessiva), ed è simmetrica.

Il linguaggio è $\mathcal{L}_{Gr} = \{E\}$, con E simbolo di relazione binario.

Un grafo è una $\mathcal{L}_{Gr}$-struttura che soddisfa i seguenti enunciati:

- $\forall x \left(\neg (xEx)\right)$;
- $\forall x \forall y (xEy \Rightarrow yEx)$

Spazio vettoriale. (3.8) Uno spazio vettoriale è una struttura algebrica avente $(V, +, \mathbf{0})$ gruppo abeliano, e con un prodotto per scalari $\mathbb{K} \times V \to V$, con $\mathbb{K}$ campo, tale che

$$\begin{aligned} a(\boldsymbol{v} + \boldsymbol{w}) &= a\boldsymbol{v} + a\boldsymbol{w} \\ (a + b)\boldsymbol{v} &= a\boldsymbol{v} + b\boldsymbol{v} \\ 1 \cdot \boldsymbol{v} &= \boldsymbol{v} \\ (ab)\boldsymbol{v} &= a(b\boldsymbol{v}) \end{aligned}$$

Questo posso formalizzarlo come:

$$\mathcal{L} = \{+, 0\} \cup \{f_a : a \in \mathbb{K}\}$$

dove f_a sono funzioni unarie. L'idea è che per la mia struttura $\mathcal{M}$:

$$\begin{aligned} f_a^{\mathcal{M}} : V &\longrightarrow V \\ \boldsymbol{v} &\longmapsto a\boldsymbol{v} \end{aligned}$$

Uno spazio vettoriale una $\mathcal{L}$-struttura $\mathcal{M}$ che soddisfa i seguenti enunciati.

Formalizzando la seconda parte (ovvero quella sui prodotti scalari), iniziamo con la associatività: scrivo un blocco di formule, una per ciascun elemento del campo

$$\begin{aligned} &\forall\, \boldsymbol{v}\, \forall\, \boldsymbol{w}\, \big(f_a(\boldsymbol{v}+\boldsymbol{w}) = f_a(\boldsymbol{v}) + f_a(\boldsymbol{w})\big) \\ &\forall\, \boldsymbol{v}\, \forall\, \boldsymbol{w}\, \big(f_b(\boldsymbol{v}+\boldsymbol{w}) = f_b(\boldsymbol{v}) + f_b(\boldsymbol{w})\big) \\ &\vdots \end{aligned}$$

Per quanto rigaurda invece la distributività, si scrive un blocco di formule, ciascuna per ogni $a, b \in \mathbb{K}$ ed ogni $c \in \mathbb{K}$ tale che $a + b = c$:

$$\begin{aligned} &\forall\, \boldsymbol{v}\, \big(f_c(\boldsymbol{v}) = f_a(\boldsymbol{v}) + f_b(\boldsymbol{v})\big) \\ &\vdots \end{aligned}$$

Spazio metrico. (3.9) Uno spazio metrico è una coppia (X, d), con $d : X \times X \longrightarrow \mathbb{R}$ tale che

1. $d(x, y) \geq 0$, $\forall\, x, y \in X$, e $d(x, y) = 0$ se e solo se $x = y$;
2. $d(x, y) = d(y, x)$, $\forall\, x, y \in X$;
3. $d(x, z) \geq d(x, y) + d(y, z)$, $\forall\, x, y, z \in X$.

Introduco un linguaggio $\mathcal{L} = \{P_q : q \in \mathbb{Q}^+\}$, dove P_q è un simbolo di relazione binaria. L'idea è che $P_q(x, y)$ sse $d(x, y) < q$. In questo contesto:

$$d(x, y) = \inf \Big\{ q \in \mathbb{Q}^+ : P_q(x, y) \Big\}$$

Dunque, uno spazio metrico è una $\mathcal{L}$-struttura che soddisfa i seguenti enunciati:

1. è una doppia implicazione:

- "$d(x,x) = 0$": scrivo una formula per ogni $q \in \mathbb{Q}^+$:

$$\forall\, x\, P_q(x,x)$$

- "se $x \neq y$, allora $d(x,y) \neq 0$" ???

2. scrivo una formula per ogni $q \in \mathbb{Q}^+$:

$$\forall\, x\, \forall\, y\, \big(P_q(x,y) \;\Leftrightarrow\; P_q(y,x)\big)$$

3.2 Interpretazione di enunciati e formule

Esempio. (3.10) Consideriamo il linguaggio $\mathcal{L} = \{R, f, c\}$, dove R è una relazione binaria, f è una funzione binaria, e c è costante. Sia σ l'enunciato:

$$\sigma: \quad \forall\, x\, \big(\neg\, \exists\, y\, f(y,y) \mathrel{\dot=} x \;\Rightarrow\; R(c,x)\big)$$

Utilizzo la struttura: $\mathcal{M} = (\mathbb{N}, <, +, 0)$:

$$\sigma^{\mathcal{M}}: \quad \forall\, x \in \mathbb{N}\, \big(\neg\, \exists\, y \in \mathbb{N}\, (y+y) = x \;\Rightarrow\; 0 < x\big)$$

σ quindi significa che "per ogni numero naturale, se è dispari allora è maggiore di 0".

Scriveremo $\mathcal{M} \vDash \sigma$ se e solo se $\sigma^{\mathcal{M}}$ è vera.

Esempio. (3.11) Consideriamo il linguaggio $\mathcal{L} = \{R, f, c\}$, dove R è una relazione binaria, f è una funzione binaria, e c è costante. Sia $\varphi(x)$ la formula:

$$\varphi(x): \quad f(x,x) \mathrel{\dot=} c$$

Utilizzo la struttura: $\mathcal{M} = (\mathbb{N}, <, +, 0)$:

$$\varphi^{\mathcal{M}}: \quad x + x = 0$$

Ci possiamo chiedere se $\mathcal{M} \vDash \varphi$? Contentendo una variabile libera (x), di suo questa affermazione non è né vera né falsa, ma dipende dal valore assegnato a x.

Interpretazione di un enunciato. (3.12) Sia $\mathcal{M}$ una $\mathcal{L}-$struttura. L'interpretazione di un $\mathcal{L}$-enunciato σ in $\mathcal{M}$ è la pseudo-formula $\sigma^{\mathcal{M}}$ ottenuta rimpiazzando ciascun simbolo $s \in \mathcal{L}$ con la sua interpretazione $s^{\mathcal{M}}$ e limitando tutte le variabili (e quindi i quantificatori) a variare sugli elementi di M.

Intepretando nella maniera usuale le costanti logiche, la pseudo-formula $\sigma^{\mathcal{M}}$ diventa un'affermazione (in linguaggio matematico) che riguarda la struttura $\mathcal{M}$. Scriveremo

$$\mathcal{M} \vDash \sigma$$

per dire che $\sigma^{\mathcal{M}}$ è un'affermazione vera in $\mathcal{M}$. In caso contrario, scriviamo $\mathcal{M} \nvDash \sigma$.

Se $\mathcal{M} \vDash \sigma$ diciamo che $\mathcal{M}$ soddisfa σ, o che $\mathcal{M}$ è un modello di σ. La relazione $\vDash$ tra strutture ed enunciati si chiama relazione di soddisfazione.

Interpretazione di formule. (3.13) Se $\varphi(x_1, \ldots, x_n)$ è invece una $\mathcal{L}$-formula contenente variabili libere, possiamo ancora definire $\varphi^{\mathcal{M}}$ come prima, ma il fatto che $\sigma^{\mathcal{M}}$ sia vera o no nella $\mathcal{L}$-struttura $\mathcal{M}$ dipenderà da quali valori (= elementi di M) assegniamo alle variabili libere.

Dati $a_1, \ldots, a_n \in M$, scriveremo $\mathcal{M} \vDash \sigma[a_1/x_1, \ldots, a_n/x_n]$ o, più brevemente,

$$\mathcal{M} \vDash \sigma[a_1, \ldots, a_n]$$

per dire che $\sigma^{\mathcal{M}}$ è vera (in $\mathcal{M}$) una volta che alle occorrenze libere di ciascuna x_i assegniamo il valore a_i. La funzione $\alpha : \{x_1, \ldots, x_n\} \longrightarrow M$ data da $\alpha(x_i) = a_i$ viene detta assegnazione e talvolta scriveremo $\mathcal{M} \vDash \varphi[\alpha]$.

Insieme di verità. (3.14) L'insieme

$$\mathbf{T}_\varphi = \mathbf{T}^{\mathcal{M}}_{\varphi(x_1,\ldots,x_n)} = \big\{(a_1, \ldots, a_n) \in M^n : \mathcal{M} \vDash \sigma[a_1, \ldots, a_n]\big\}$$

è detto insieme di verità di φ in $\mathcal{M}$.

3.3 Interpretazione nella Logica proposizionale

Logica proposizionale. (3.15) Sia S un insieme di lettere proposizionali. L'insieme $\mathrm{Prop}(S)$ delle proposizioni (o formula proposizionali) su S è dato da tutte le stringhe che possono essere costruite a partire dagli elementi di S usando i connettivi come descritto nella logica del prim'ordine, ovvero:

- A è una proposizione per ogni $A \in S$;
- se P è una proposizione, allora lo è anche $(\neg P)$;
- se P e Q sono proposizioni e $\odot$ è uno tra $\wedge, \vee, \Longrightarrow, \Longleftrightarrow$ allora anche $(P \odot Q)$ è una proposizione.

Scriviamo $P(A_1, \dots, A_n)$ per dire che le lettere proposizionali che occorrono in $\mathcal{P}$ sono tra le $A_1, \dots, A_n$. Se $Q_1, \dots, Q_n \in \mathrm{Prop}(S)$

$$P\left[Q_1/A_1, \dots, Q_n/A_n\right]$$

è la proposizione che si ottiene sostituendo ciascun A_i con Q_i.

Si applicano le solite convenzioni per eliminare le parentesi non necessarie.

Valutazione. (3.16) Ricordando come è stata definita la funzione valutazione[†], possiamo associare ad ogni proposizione $P(A_1, \dots, A_n)$ una funzione

$$f_P : \{0,1\}^n \longrightarrow \{0,1\}$$

tale che $f_P(i_1, \dots, i_n) = 1$ se e solo se $\overline{v}(P) = i$ per qualche/ogni valutazione v tale che $v(A_k) = i_k$ per $1 \leq k \leq n$.

La tabella che riporta il grafico di f_P si chiama tavola di verità di P.

Se R è la proposizione $P[Q_1/A_1, \dots, Q_n/A_n]$ e $Q_k(B_1, \dots, B_m)$ per ogni $1 \leq k \leq n$ allora

$$f_R(i_1, \dots, i_m) = f_P\left(f_{Q_1}(i_1, \dots, i_m), \dots, f_{Q_n}(i_1, \dots, i_m)\right).$$

Esempio. (3.17) Data una proposizione

$$P: \quad (\neg A \wedge B) \Rightarrow C$$

costruisco la tavola di verità:

[†] Vedi (1.7)

A	B	C	$\neg A$	$\neg A \wedge B$	P
0	0	0	1	0	1
0	0	1	1	0	1
0	1	0	1	1	0
0	1	1	1	1	1
1	0	0	0	0	1
1	0	1	0	0	1
1	1	0	0	0	1
1	1	1	0	0	1

Esempio. (3.18) Consideriamo una proposizione:

$$Q: \quad (A \Rightarrow B) \vee (B \Rightarrow A)$$

la cui tavola di verità è

A	B	$A \Rightarrow B$	$B \Rightarrow A$	Q
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	1	1	1

Osservo che, a differenza dell'esempio precedente, Q è sempre vera, a prescindere dal valore di verità di A e B.

Esempio. (3.19) Consideriamo una proposizione:

$$R: \quad A \vee \neg A$$

la cui tavola di verità è

A	$\neg A$	R
0	1	0
1	0	0

Definizione. (3.20) *Una proposizione P è una tautologia (in simboli $\vDash P$) se $v(P) = 1$ per ogni valutazione v, ed p una contraddizione proposizionale se $v(P) = 0$ per ogni v.*

Osservazione. (3.21) Se $P(A_1, \ldots, A_n)$ è una tautologia e $Q_1, \ldots, Q_n \in \mathrm{Prop}(S)$, allora anche $P[Q_1/A_1, \ldots, Q_n/A_n]$ è una tautologia.

Osservazione. (3.22) P è una tautologia se e solo se $\neg P$ è una contraddizione.

Esempio. (3.23) Consideriamo le due proposizioni:

$$P: \quad \neg A \wedge B, \qquad Q: \quad A \vee B$$

le cui tavole di verità sono:

A	B	$\neg A$	P	Q
0	0	1	0	0
0	1	1	1	1
1	0	0	1	1
1	1	0	1	1

Definizione. (3.24) *Date due proposizioni P e Q, diciamo che P è tautologicamente equivalente a Q (in simboli $P \equiv Q$) se $v(P) = v(Q)$ per ogni valutazione v.*

Osservazione. (3.25) P e Q sono tautologicamente equivalenti se e solo se $P \Leftrightarrow Q$ è una tautologia

Esempio. (3.26) Consideriamo le due proposizioni:

$$P: \quad A \vee B, \qquad Q: \quad A \Rightarrow B$$

le cui tavole di verità sono:

A	B	P	Q
0	0	0	1
0	1	0	1
1	0	0	0
1	1	1	1

Definizione. (3.27) *Una proposizione P è conseguenza tautologica di un insieme di proposizioni Γ (in simboli $\Gamma \vDash P$) se $v(P) = 1$ per ogni valutazione v tale che $v(Q) = 1$ per ogni $Q \in \Gamma$.*

Osservazione. (3.28) Q è conseguenza tautologica di P se e solo se $P \Rightarrow Q$ è tautologia

Inoltre, se $\Gamma = \{Q_1, \ldots, Q_n\}$ è finito, allora $\Gamma \models P$ se e solo se

$$(Q_1 \wedge \ldots \wedge Q_n) \Rightarrow P$$

è una tautologia

3.3.1 Applicazione alla logica del prim'ordine

Intepretazione delle formule del prim'ordine. (3.29) Ogni formula del prim'ordine si scrive in maniera unica come combinazione booleana di formule primitive, ovvero è un elemento di $\mathrm{Prop}(S)$ dove S è l'insieme delle formule atomiche e delle formule esistenziali o universali (= formule primitive). In altre parole, σ è nella forma

$$P_\sigma[\psi_1/A_1, \ldots, \psi_n/A_n]$$

con P_σ formula proposizionale e $\psi_1, \ldots, \psi_n$ formule primitive.

Definizione. (3.30) *Un $\mathcal{L}$-enunciato σ è una <u>tautologia</u> se tale è la corrisponente formula proposizionale P_σ.*

Osservazione. (3.31) Chiaramente, se σ è una tautologia, allora $\mathcal{M} \models \sigma$ per ogni $\mathcal{L}$-struttura $\mathcal{M}$.

Estensioni. (3.32) Il concetto di tautologia si può estendere, con la stessa esatta definizione, a $\mathcal{L}$-formule arbitrarie: in questo caso, se $\sigma(x_1, \ldots, x_n)$ è una tautologia, allora $\mathcal{M} \models \sigma[a_1, \ldots, a_n]$ per ogni $\mathcal{L}$-struttura $\mathcal{M}$ ed ogni $a_1, \ldots, a_n \in M$.

Esempio. (3.33) Considero la formula:

$$\forall x \, \exists y \, P(x,y) \wedge \forall z \left(P(x,z) \Rightarrow z \doteq x\right)$$

chiamo

$$\begin{aligned} A &\coloneqq \forall x \, \exists y \, P(x,y) \\ B &\coloneqq \forall z \left(P(x,z) \Rightarrow z \doteq x\right) \end{aligned}$$

e questa formula diventa, nella logica proposizionale, $A \wedge B$ (che non è una tautologia).

Perché ho dovuto scegliere proprio quelle come A e B? Costruiamone l'albero sintattico:

$$
\begin{tikzcd}[column sep=small]
 & \forall x\, \exists y\, P(x,y) \wedge \forall z \big(P(x,z) \Rightarrow z = x\big) \arrow[dl, no head] \arrow[drr, no head] & & & \\
\boxed{\forall x\, \exists y\, P(x,y)} \arrow[d, no head] & & & \boxed{\forall z \big(P(x,z) \Rightarrow z = x\big)} \arrow[d, no head] & \\
\exists y\, P(x,y) \arrow[d, no head] & & & P(x,z) \Rightarrow z = x \arrow[dl, no head] \arrow[dr, no head] & \\
P(x,y) & & P(x,z) & & z = x
\end{tikzcd}
$$

Osservo quindi che, per scegliere le formule primitive di una formula del primo ordine, scendo lungo l'albero sintattico finché non aggiungo un quantificatore, e mi fermo alla sottoformula immediatamente più in alto.

3.4 Interpretazione dei termini

Interpretazione. (3.34) Sia t un $\mathcal{L}$-termine e $x_1, \dots, x_n$ le variabili che occorrono in t. Sia $\mathcal{M}$ una $\mathcal{L}$-struttura e $a_1, \dots, a_n \in M$. Allora

$$t^{\mathcal{M}}[a_1, \dots, a_n]$$

è l'elemento di M definito per ricorsione da

- se t è nella forma x_i, allora $t^{\mathcal{M}}[a_1, \dots, a_n] = a_i$;
- se t è nella forma c per qualche $c \in \mathrm{Cost}$, allora $t^{\mathcal{M}}[a_1, \dots, a_n] = c^{\mathcal{M}}$;
- se t è nella forma $f(s_1, \dots, s_k)$ per qualche $f \in \mathrm{Fun}$ con $\mathrm{ar}(f) = k$ e $s_1, \dots, s_k$ termini, allora
$$t^{\mathcal{M}}[a_1, \dots, a_n] = f^{\mathcal{M}} \left(s_1^{\mathcal{M}}[a_1, \dots, a_n], \dots, s_k^{\mathcal{M}}[a_1, \dots, a_n]\right)$$

In altre parole, $t^{\mathcal{M}}$ è la funzione n-aria

$$t^{\mathcal{M}} : M^n \longrightarrow M$$

orttenuta rimpiazzando ciascun $f \in \mathrm{Fun}$ e $c \in \mathrm{Cost}$ con $f^{\mathcal{M}}$ e $c^{\mathcal{M}}$, rispettivamente, e $t^{\mathcal{M}}$ è il valore di $t^{\mathcal{M}}$ sull'input $(a_1, \dots, a_n)$.

Esempio. (3.35) Sia $\mathcal{L} = \{f, g, h, c\}$, f, g funzioni binarie, h funzione unaria e c costante.

Consideriamo il termine

$$t(x, y): \quad f\Big(g\big(h(c), x\big), h(y)\Big)$$

e la struttura:

$$\mathcal{M} = (\mathbb{N}, +, \cdot, \exp_2, 1).$$

- $t^{\mathcal{M}} : (2^1 \cdot x) + 2^y$.
- Alcuni esempi di assegnazioni sono:

$$t^{\mathcal{M}}[2, 3] = 2 \cdot 2 + 2^3 = 12;$$
$$t^{\mathcal{M}}[1, 0] = 2 \cdot 1 + 2^0 = 3$$

Se cambio struttura, e scelgo

$$\mathcal{N} = (\mathbb{Z}, \cdot, +, -, 0).$$

- $t^{\mathcal{N}} : \big((-0) + x\big) \cdot (-y) \rightsquigarrow t^{\mathcal{N}} : -x \cdot y$.
- Alcuni esempi di assegnazioni sono:

$$t^{\mathcal{N}}[-2, 5] = -\big((-2) \cdot 5\big) = 10$$
$$t^{\mathcal{N}}[5, -2] = -\big(5 \cdot (-2)\big) = 10$$
$$t^{\mathcal{N}}[0, 0] = 0$$

Osservazione. (3.36) I termini, utilizzati in questo modo, possono servire per utilizzare nuovi elementi all'interno della struttura; nell'esempio precedente:

$$t^{\mathcal{M}}[1, 1] = 4$$

posso quindi definire un termine $t' = t[c/x, c/x]$ che è esattamente il numero naturale 4.

3.5 Validità delle formule

Relazione di soddisfazione. (3.37) Data una $\mathcal{L}$-struttura $\mathcal{M}$ ed una formula $\varphi(x_1, \ldots, x_n)$, definiamo la relazione di soddisfazione

$$\mathcal{M} \vDash \varphi[a_1, \ldots, a_n]$$

per ricorcorsione sulla complessità di $\varphi(x_1, \ldots, x_n)$.

- Se $\varphi(x_1, \dots, x_n)$ è del tipo

$$\varphi(x_1, \dots, x_n): \qquad t \doteq s$$

(con t ed s termini), allora $\mathcal{M} \vDash \sigma[a_1, \dots, a_n]$ se e solo se

$$t^{\mathcal{M}}[a_1, \dots, a_n] = s^{\mathcal{M}}[a_1, \dots, a_n].$$

- Se $\varphi(x_1, \dots, x_n)$ è del tipo

$$\varphi(x_1, \dots, x_n): \qquad R(t_1, \dots, t_k)$$

per R relazione e t_i termini, allora $\mathcal{M} \vDash \varphi[a_1, \dots, a_n]$ se e solo se

$$\left(t_1^{\mathcal{M}}[a_1, \dots, a_n], \dots, t_k^{\mathcal{M}}[a_1, \dots, a_n]\right) \in R^{\mathcal{M}}.$$

- Se $\varphi(x_1, \dots, x_n)$ è del tipo

$$\varphi(x_1, \dots, x_n): \qquad \neg\,\psi$$

allora $\mathcal{M} \vDash \varphi[a_1, \dots, a_n]$ se e solo se

$$\mathcal{M} \nvDash \psi[a_1, \dots, a_n].$$

- Se $\varphi(x_1, \dots, x_n)$ è del tipo

$$\psi_1 \,\wedge\, \psi_2$$

allora $\mathcal{M} \vDash \varphi[a_1, \dots, a_n]$ se e solo se

$$\mathcal{M} \vDash \psi_1[a_1, \dots, a_n] \quad \text{e} \quad \mathcal{M} \vDash \psi_2[a_1, \dots, a_n].$$

- Se $\varphi(x_1, \dots, x_n)$ è del tipo

$$\psi_1 \,\vee\, \psi_2$$

allora $\mathcal{M} \vDash \varphi[a_1, \dots, a_n]$ se e solo se

$$\mathcal{M} \vDash \psi_1[a_1, \dots, a_n] \quad \text{oppure (inclusivo)} \quad \mathcal{M} \vDash \psi_2[a_1, \dots, a_n].$$

- Se $\varphi(x_1, \ldots, x_n)$ è del tipo
$$\psi_1 \Rightarrow \psi_2$$
allora $\mathcal{M} \vDash \varphi[a_1, \ldots, a_n]$ se e solo se
$$\mathcal{M} \nvDash \psi_1[a_1, \ldots, a_n] \quad \text{oppure} \quad \mathcal{M} \vDash \psi_2[a_1, \ldots, a_n].$$

- Se $\varphi(x_1, \ldots, x_n)$ è del tipo
$$\psi_1 \Leftrightarrow \psi_2$$
allora $\mathcal{M} \vDash \varphi[a_1, \ldots, a_n]$ se e solo se
$$\mathcal{M} \vDash \psi_1[a_1, \ldots, a_n] \quad \text{se e solo se} \quad \mathcal{M} \vDash \psi_2[a_1, \ldots, a_n].$$

- Se $\varphi(x_1, \ldots, x_n)$ è del tipo $\exists\, y\, \psi$ dobbiamo distinguere tre casi.
  - Se y non occorre libera in ψ, allora possiamo scrivere ψ come $\psi(x_1, \ldots, x_n)$ e $\mathcal{M} \vDash \psi[a_1, \ldots, a_n]$ se e solo se
  $$\mathcal{M} \vDash \psi[a_1, \ldots, a_n].$$
  - Se y occorre libera in ψ ed è diversa da tutte le x_i, allora possiamo scrivere ψ come $\psi(x_1, \ldots, x_n, y)$, e $\mathcal{M} \vDash \varphi[a_1, \ldots, a_n]$ se e solo se esiste qualche $b \in M$ tale che
  $$\mathcal{M} \vDash \psi[a_1, \ldots, a_n, b].$$
  - Se y occorre libera in ψ è $y = x_i$ per qualche i, allora possiamo scrivere ψ come $\psi(x_1, \ldots, x_{i-1}, y, x_{i+1}, \ldots, x_n)$ e si avrò che $\mathcal{M} \vDash \varphi[a_1, \ldots, a_m]$ se e solo se esiste qualche $b \in M$ tale che
  $$\mathcal{M} \vDash \psi[a_1, \ldots, a_{i-1}, b, a_{i+1}, \ldots, a_n].$$

- Se $\varphi(x_1, \ldots, x_n)$ è del tipo $\forall\, y\, \psi$ dobbiamo distinguere tre casi.
  - Se y non occorre libera in ψ, allora possiamo scrivere ψ come $\psi(x_1, \ldots, x_n)$ e $\mathcal{M} \vDash \psi[a_1, \ldots, a_n]$ se e solo se
  $$\mathcal{M} \vDash \psi[a_1, \ldots, a_n].$$

– Se y occorre libera in ψ ed è diversa da tutte le x_i, allora possiamo scrivere ψ come $\psi(x_1, \dots, x_n, y)$, e $\mathcal{M} \vDash \varphi[a_1, \dots, a_n]$ se e solo se per ogni $b \in M$ si ha che

$$\mathcal{M} \vDash \psi[a_1, \dots, a_n, b].$$

– Se y occorre libera in ψ è $y = x_i$ per qualche i, allora possiamo scrivere ψ come $\psi(x_1, \dots, x_{i-1}, y, x_{i+1}, \dots, x_n)$ e si avrò che $\mathcal{M} \vDash \varphi[a_1, \dots, a_m]$ se e solo se per ogni $b \in M$ si ha che

$$\mathcal{M} \vDash \psi[a_1, \dots, a_{i-1}, b, a_{i+1}, \dots, a_n].$$

Esempio. (3.38) Consideriamo:

$$\mathcal{L} = \{R, f, g, h, c\}$$
$$\mathcal{M} = (\mathbb{N}, \leq, +, \cdot, \exp_2, 1)$$

e i termini

$$t(x, y) : \quad f\left(h(c), x\right) \qquad s(x, y) : \quad g(x, y).$$

Consideriamo anche la formula

$$\varphi(x, y) : \quad t \doteq s.$$

Ci chiediamo se $\mathcal{M} \vDash \varphi[0, 2]$.

Iniziamo valutando $t^{\mathcal{M}}$ e $s^{\mathcal{M}}$:

$$t^{\mathcal{M}} : \qquad 2^1 + x$$
$$s^{\mathcal{K}} : \qquad x \cdot y$$

per poi assegnare i valori:

$$t^{\mathcal{M}}[0, 2] = 0 + 2 = 2$$
$$s^{\mathcal{M}}[0, 2] = 0 \cdot 2 = 0$$

Quindi si ha che $\mathcal{M} \vDash \varphi[0, 2]$ se e solo se $t^{\mathcal{M}}[0, 2] = s^{\mathcal{M}}[0, 2]$, ovvero se e solo se $2 = 0$. FALSO

Esempio. (3.39) Consideriamo:

$$\begin{aligned} \mathcal{L} &= \{R, f, g, h, c\} \\ \mathcal{M} &= (\mathbb{N}, \leq, +, \cdot, \exp_2, 1) \end{aligned}$$

e la formula:

$$\varphi : \quad \neg \forall x \left(R(x, c) \Rightarrow \exists z \left(h(z) \doteq x \right) \right)$$

Ci chiediamo se $\mathcal{M} \vDash \varphi$.

$$\begin{array}{c} \neg \forall x \left(R(x, c) \Rightarrow \exists z \left(h(z) \doteq x \right) \right) \\ | \\ \forall x \left(R(x, c) \Rightarrow \exists z \left(h(z) \doteq x \right) \right) \\ | \\ R(x, c) \Rightarrow \exists z \left(h(z) \doteq x \right) \\ \swarrow \qquad \searrow \\ R(x, c) \qquad \exists z \left(h(z) \doteq x \right) \\ \qquad\qquad | \\ \qquad\qquad h(z) \doteq x \end{array}$$

Essenzialmente, $\varphi^{\mathcal{M}}$ è

> Non è vero che per ogni $b \in \mathbb{N}$, $b \not\leq 1$ oppure per qualche $d \in \mathbb{N}$, $2^d = b$

Formule con variabili libere. (3.40) Il fatto che $\mathcal{M} \vDash \varphi[a_1, \ldots, a_n]$ dipende solo dalle variabili $x_{i_1}, \ldots, x_{i_k}$ che davvero occorono libere in ψ:

$$\mathcal{M} \vDash \varphi[a_1, \ldots, a_n] \quad \text{se e solo se} \quad \mathcal{M} \vDash \varphi[a_{i_1}, \ldots, a_{i_k}]$$

dove a sinistra pensiamo φ come $\varphi(x_1, \ldots, x_n)$ e a destra la pensiamo come $\varphi(x_{i_1}, \ldots, x_{i_k})$.

Quindi se φ è un enunciato possiamo legittimamente scrivere

$$\mathcal{M} \vDash \varphi.$$

Chiusura universale. (3.41) Per convenzione, se φ contiene le variabili libere $x_1, \dots, x_n$ si scrive

$$\mathcal{M} \vDash \varphi$$

per dire che $\mathcal{M}$ soddisfa la chiusura universale $\varphi^\forall$ di φ.

Interpretazione insiemistica. (3.42) Osserviamo che date $\varphi(x_1, \dots, x_n)$ e $\psi(x_1, \dots, x_n)$ si ha

$$\begin{aligned} \mathbf{T}_{\neg\varphi} &= M^n \setminus \mathbf{T}_\varphi \\ \mathbf{T}_{\varphi\wedge\psi} &= \mathbf{T}_\varphi \cap \mathbf{T}_\psi \\ \mathbf{T}_{\varphi\vee\psi} &= \mathbf{T}_\varphi \cup \mathbf{T}_\psi. \end{aligned}$$

Inoltre, se $\varphi(x_1, \dots, x_n)$ è $\exists\, y\, \varphi(x_1, \dots, x_n, y)$ con $y \neq x_i$, allora

$$\mathbf{T}_{\varphi(x_1,\dots,x_n)} = p\left[\mathbf{T}_{\psi(x_1,\dots,x_n,y)}\right]$$

dove $p : M^{n+1} \longrightarrow M^n$ è la proiezione sulle prime n coordinate, ovvero

$$p(a_1, \dots, a_n, y) = (a_1, \dots, a_n).$$

Gli altri connettivi e il quantificatore universale danno luogo ad operazioni insiemistiche "derivate". Ad esempio

$$\mathbf{T}_{\varphi\Rightarrow\psi} = (M^n \setminus \mathbf{T}_\varphi) \cup \mathbf{T}_\psi.$$

Data una formula $\varphi(x_1, \dots, x_n)$

$$\begin{aligned} &\mathcal{M} \vDash \exists\, x_1 \dots \exists\, x_n\, \varphi \quad \text{se e solo se} \quad \mathbf{T}_{\varphi(x_1,\dots,x_n)} \neq \emptyset \\ &\mathcal{M} \vDash \forall\, x_1 \dots \forall\, x_n\, \varphi \quad \text{se e solo se} \quad \mathbf{T}_{\varphi(x_1,\dots,x_n)} = M^n \end{aligned}$$

Terminologia. (3.43) Una $\mathcal{L}$-formula φ si dice:

- valida se $\mathcal{M} \vDash \varphi$ per ogni $\mathcal{L}$-struttura $\mathcal{M}$; osserviamo che non è uguale alla tautologia: quest'ultima, infatti, è tale per motivi proposizionali; quindi "tautologia" $\Rightarrow$ "formula valida";
- insoddisfacibile se non esiste alcuna $\mathcal{L}$-struttura $\mathcal{M}$ tale che $\mathcal{M} \vDash \varphi$;
- soddisfacibile se esiste una $\mathcal{L}$-struttura $\mathcal{M}$ tale che $\mathcal{M} \vDash \varphi$.

Notazione. (3.44) Per indicare che una $\mathcal{L}$-formula è valida scriviamo

$$\models \varphi.$$

Osservazione. (3.45) Chiaramente se φ è valida allora è anche soddisfacibile, ed un enunciato σ è valido se e solo se $\neg\,\sigma$ è insoddisfacibile

Notazione. (3.46) Dato un insieme Σ di $\mathcal{L}$-formule ed un $\mathcal{L}$-struttura $\mathcal{M}$, scriviamo $\mathcal{M} \models \Sigma$ quanto $\mathcal{M} \models \varphi$ èer ogni $\varphi \in \Sigma$.

Definizione. (3.47) *Una $\mathcal{L}$-formula è <u>conseguenza logica</u> di un insieme Σ di $\mathcal{L}$-formule, in simboli*

$$\Sigma \models \sigma$$

se per ogni $\mathcal{L}$-struttura $\mathcal{M}$ tale che $\mathcal{M} \models \Sigma$ vale anche $\mathcal{M} \models \varphi$.

Definizione. (3.48) *Due $\mathcal{L}$-formule φ e ψ sono <u>logicamente equivalenti</u>, in simboli*

$$\varphi \equiv \psi$$

se $\varphi \models \psi$ e $\psi \models \varphi$, ovvero se per ogni $\mathcal{L}$-struttura $\mathcal{M}$ si ha che $\mathcal{M} \models \varphi$ se e solo se $\mathcal{M} \models \psi$.

Osservazione. (3.49) Se $\Sigma = \{\psi_1, \dots, \psi_n\}$ è finito omettiamo le parentesi graffe e scriviamo $\psi_1, \dots, \psi_n \models \varphi$. In questo caso $\Sigma \models \varphi$ se e solo se $(\psi_1 \wedge \ldots \wedge \psi_n) \implies \varphi$ è valida. Similmente $\varphi \equiv \psi$ se e solo se $\varphi \Leftrightarrow \psi$ è valida

Osservazione. (3.50) Se $\Sigma = \{\varphi_1, \dots, \varphi_n\}$ insieme <u>finito</u> di $\mathcal{L}$-formule, con $\mathcal{M}$ $\mathcal{L}$-struttura, si ha che

$$\begin{array}{lcr} \mathcal{M} \models \Sigma & \text{sse} & \mathcal{M} \models \varphi_i \text{ per ogni } i \\ & \text{sse} & \mathcal{M} \models \varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n \end{array}$$

Usi di $\models$. (3.51) Se φ è un enunciato o una formula, $\mathcal{M}$ è una $\mathcal{L}$-struttura e Σ è un insieme di $\mathcal{L}$-formule:

1. $\mathcal{M} \models \varphi$ è una relazione di soddisfazione;
2. $\models \varphi$ rappresenta la validità:
3. $\Sigma \models \varphi$ è una relazione di <u>conseguenza logica</u>.

Osservazione. (3.52) Vale che

$$\emptyset \models \varphi \qquad \text{sse} \qquad \models \varphi.$$

Osservazione. (3.53) Vale che

$$\varphi \models \psi \qquad \text{sse} \qquad \models \varphi \Rightarrow \psi$$

Capitolo 4

$\mathcal{L}$-struttura

4.1 $\mathcal{L}$-teoria

Definizione. (4.1)

- *Una* <u>*$\mathcal{L}$-teoria*</u> *è un insieme T di $\mathcal{L}$-enunciati, ed $\mathcal{L}$ si dice* <u>*lingauggio di T*</u>*. Una teoria del prim'ordine è una $\mathcal{L}$-teoria per qualche linguaggio del prim'ordine $\mathcal{L}$.*
- *Un* <u>*sistema di assiomi*</u> *(o assiomatizzazione) di una $\mathcal{L}$-teoria T è un insieme di $\mathcal{L}$-enunciati Σ tale che per ogni $\mathcal{L}$-enunciato σ*

$$T \vDash \sigma \quad \textit{se e solo se} \quad \Sigma \vDash \sigma$$

Osservazione. (4.2) Affinché $\Sigma \subseteq T$ sia un'assiomatizzazione di T è sufficiente che $\Sigma \vDash \sigma$ per ogni $\sigma \in T$.

Definizione. (4.3) *Sia $\mathcal{C}$ una collezione di $\mathcal{L}$-strutture. La* <u>*teoria di $\mathcal{C}$*</u> *è l'insieme* $\mathrm{Th}(\mathcal{C})$ *di tutti gli $\mathcal{L}$-enunciati σ tali che $\mathcal{M} \vDash \sigma$ per ogni $\mathcal{M} \in \mathcal{C}$. Se $\mathcal{C} = \{\mathcal{M}\}$ scriviamo* $\mathrm{Th}(\mathcal{M})$ *anziché* $\mathrm{Th}\big(\{\mathcal{M}\}\big)$.

Notazione. (4.4) Data una $\mathcal{L}$-teoria T, indichiamo con

$$\mathrm{Mod}(T)$$

l'insieme dei modelli di T (ovvero delle $\mathcal{L}$-strutture $\mathcal{M}$ tali che $\mathcal{M} \vDash T$).

Quando $T = \{\sigma\}$ scriviamo $\mathrm{Mod}(\sigma)$ al posto di $\mathrm{Mod}\big(\{\sigma\}\big)$.

Osservazione. (4.5) Vale che

$$\mathrm{Mod}(T_0 \cup T_1) = \mathrm{Mod}(T_0) \cap \mathrm{Mod}(T_1)$$

e inoltre, se $T_0 \subseteq T_1$, si ha che

$$\mathrm{Mod}(T_1) \subseteq \mathrm{Mod}(T_0).$$

Definizione. (4.6) *Una $\mathcal{L}$-teoria T si dice <u>soddisfacibile</u> se ha un modello, ovvero se* $\mathrm{Mod}(T) \neq \emptyset$*; in caso contrario T è <u>insoddisfacibile</u>.*

Esempio. (4.7) Una teoria <u>insoddisfacibile</u> potrebbe essere:

$$T = \{\exists\, x\, \neg(x \doteq x)\}$$

in quanto, per definizione, in ogni linguaggio e struttura, $x \doteq x$, e dunque

$$\mathrm{Mod}(T) = \emptyset$$

Proposizione. (4.8) Sia $\Sigma \cup \{\sigma\}$ un insieme di $\mathcal{L}-$enunciati. Allora

$$\Sigma \models \sigma \text{ se e solo se } \Sigma \cup \{\neg\, \sigma\} \text{ è insoddisfacibile.}$$

Dimostrazione di (4.8)

$\boxed{\Rightarrow}$ Sia $\mathcal{M}$ tale che $\mathcal{M} \models \Sigma$.

Allora $\mathcal{M} \models \sigma$, perciò $\mathcal{M} \models \neg\, \sigma$.

Perciò, $\mathcal{M} \not\models \Sigma \cup \{\neg\, \sigma\}$, e poiché $\mathcal{M}$ è arbitrario, $\Sigma \cup \{\neg\, \sigma\}$ è insoddisfacibile.

$\boxed{\Longleftarrow}$ Sia $\mathcal{M}$ tale che $\mathcal{M} \models \Sigma$.

Poiché $\Sigma \cup \{\neg\, \sigma\}$ è insoddisfacibile, $\mathcal{M} \not\models \neg\, \sigma$, quindi $\mathcal{M} \models \sigma$.

Poiché $\mathcal{M}$ è arbitrario, $\Sigma \models \sigma$. ■

Definizione. (4.9) *Una $\mathcal{L}$-teoria si dice completa se è soddisfacibile e per ogni $\mathcal{L}$-enunciato σ si ha che*

$$T \models \sigma \quad \textit{oppure} \quad T \models \neg\, \sigma.$$

Osservazione. (4.10) La completezza non è una caratteristica fondamentale: ci sono casi in cui si vuole che una certa teoria sia completa, e altri casi per cui è meglio che una teoria non lo sia. I due esempi che seguono ne sono una prova.

Esempio. (4.11) Sia $T = \mathrm{Th}(\mathcal{C})$, dove $\mathcal{C}$ è l'insieme di tutti i gruppi, nel linguaggio

$$\mathcal{L} = \{*, f, e\}.$$

Se consideriamo l'enunciato

$$\sigma : \quad \forall\, x \forall\, y\, (x * y \doteq y * x)$$

ci chiediamo se $T \models \sigma$; questo è vero se e solo se ogni $\mathcal{M} \in \mathcal{C}$ soddisfa σ. Poiché ci sono i gruppi non abeliani, $T \not\models \sigma$.

È allo stesso modo evidente che $T \not\models \neg\, \sigma$, poiché ci sono gruppi abeliani.

Ne consegue che una teoria "interessante" come la teoria dei gruppi non è completa.

Esempio. (4.12) Consideriamo $\mathrm{Th}(\mathcal{N})$, dove $\mathcal{N} = (\mathbb{N}, +, \cdot, 0, 1)$.

A inizio novecento questa teoria è stata oggetto di una domanda: esiste un sistema di assiomi Σ (finito?) per $\mathrm{Th}(\mathcal{N})$?

Vorremmo che:

1. $\mathcal{N} \models \Sigma$;
2. Σ è completa (cioè decide tutte le congetture).

Definizione. (4.13) *Due $\mathcal{L}$-strutture $\mathcal{M}$ e $\mathcal{N}$ si dicono elementarmente equivalenti se* $\mathrm{Th}(\mathcal{M}) = \mathrm{Th}(\mathcal{N})$.

Teorema II.

Sia T una $\mathcal{L}-$teoria soddisfacibile. Sono fatti equivalenti:

1. T è completa;
2. T è un sistema di assiomi di $\mathrm{Th}(\mathcal{M})$ per qualche $\mathcal{L}-$struttura $\mathcal{M}$;
3. per ogni $\mathcal{M}, \mathcal{N} \in \mathrm{Mod}(T)$ si ha che $\mathcal{M}$ e $\mathcal{N}$ sono elementarmente equivalenti.

Lemma. (4.14) Notiamo che

$$\mathrm{Th}(\mathcal{M}) \vDash \sigma \quad \text{se e solo se} \quad \sigma \in \mathrm{Th}(\mathcal{M}).$$

Dimostrazione di (4.14)

$\boxed{\Longleftarrow}$ Ovvio.

$\boxed{\Rightarrow}$ Siccome $\mathcal{M} \vDash \mathrm{Th}(\mathcal{M})$ per definizione, e $\mathrm{Th}(\mathcal{M}) \vDash \sigma$, $\mathcal{M} \vDash \sigma$ e quindi, per definizione, $\sigma \in \mathrm{Th}(\mathcal{M})$. ■

Dimostrazione di II. $\boxed{1. \Rightarrow 2.}$ Sia $\mathcal{M}$ tale che $\mathcal{M} \vDash T$ (poiché T è soddisfacibile).

Per un generico σ, vale che

$$\mathrm{Th}(\mathcal{M}) \vDash \sigma \quad \underset{\dagger}{\text{sse}} \quad \sigma \in \mathrm{Th}(\mathcal{M}) \quad \underset{\ddagger}{\text{sse}} \quad \mathcal{M} \vDash \sigma \tag{4.1}$$

Voglio provare che T è un sistema di assiomi per $\mathrm{Th}(\mathcal{M})$, cioè per ogni σ

$$T \vDash \sigma \quad \text{sse} \quad \mathrm{Th}(\mathcal{M}) \vDash \sigma$$

† Per il lemma precedente.
‡ Per definizione.

ovvero
$$T \vDash \sigma \quad \text{sse} \quad \mathcal{M} \vDash \sigma$$

Se $T \vDash \sigma$, poiché $\mathcal{M} \vDash T$, allora $\mathcal{M} \vDash \sigma$.

Viceversa, se $\mathcal{M} \vDash \sigma$, allora $\mathcal{M} \nvDash \neg\sigma$, e quindi $T \nvDash \neg\sigma$. Siccome T è completa, allora $T \vDash \sigma$.

$\boxed{2. \Rightarrow 3.}$ Dimostro che per ogni $\mathcal{N} \vDash T$, $\mathrm{Th}(\mathcal{N}) = \mathrm{Th}(\mathcal{M})$, dove $\mathcal{M}$ è una struttura tale per cui T è un sistema di assiomi per $\mathrm{Th}(\mathcal{M})$.

Dato σ ho due casi:

- se $\mathcal{M} \vDash \sigma$, allora $\mathrm{Th}(\mathcal{M}) \vDash \sigma$ (per 4.1), e quindi per ipotesi, $T \vDash \sigma$; poiché $\mathcal{N} \vDash T$, si ha $\mathcal{N} \vDash \sigma$. Segue che $\mathrm{Th}(\mathcal{M}) \subseteq \mathrm{Th}(\mathcal{N})$;
- se $\mathcal{M} \nvDash \sigma$, allora $\mathcal{M} \vDash \neg\sigma$, quindi come prima ottengo $\mathcal{N} \vDash \neg\sigma$, da cui $\mathcal{N} \nvDash \sigma$. Segue che $\mathrm{Th}(\mathcal{N}) \subseteq \mathrm{Th}(\mathcal{M})$.

Quindi $\mathrm{Th}(\mathcal{M}) = \mathrm{Th}(\mathcal{N})$

$\boxed{3. \Rightarrow 1.}$ Dimostro il contrappositivo, ovvero che se non vale 1., allora non vale 3..

Se non vale 1. allora esiste un enunciato σ tale che
$$T \nvDash \sigma \quad \text{e} \quad T \nvDash \neg\sigma.$$

- $T \nvDash \sigma$ significa che esiste $\mathcal{M}$ tale che $\mathcal{M} \vDash T$, ma $\mathcal{M} \nvDash \sigma$, ovvero
$$\mathcal{M} \vDash T, \quad \text{e} \quad \mathcal{M} \vDash \neg\sigma;$$
- $T \nvDash \neg\sigma$ significa che esiste $\mathcal{N}$ tale che $\mathcal{N} \vDash T$, ma $\mathcal{N} \nvDash \neg\sigma$.

Quindi $\mathcal{M}, \mathcal{N} \in \mathrm{Mod}(T)$, ma non sono elementarmente equivalenti, come testimoniato da $\neg\sigma$. ■

4.2 Sottostrutture

Definizione. (4.15) *Sia $\mathcal{M}$ una $\mathcal{L}$-struttura con dominio M. Una <u>sottostruttura</u> di $\mathcal{M}$ è una $\mathcal{L}$-struttura $\mathcal{N}$ con dominio $N \subseteq M$ tale che*

- $R^{\mathcal{N}} = R^{\mathcal{M}} \cap N^k$ *per ogni* $R \in \mathrm{Rel}$ *con* $\mathrm{ar}(R) = k$;
- $f^{\mathcal{N}} = f^{\mathcal{M}}|_{N^k}$ *per ogni* $f \in \mathrm{Fun}$ *con* $\mathrm{ar}(f) = k$;

- $c^{\mathcal{N}} = c^{\mathcal{M}}$ *per ogni* $c \in \mathrm{Cost}$.

Notazione. (4.16) Scriviamo $\mathcal{N} \subseteq \mathcal{M}$ per dire che $\mathcal{N}$ è una sottostruttura di $\mathcal{M}$.

Osservazione. (4.17) Dato $\emptyset \neq D \subseteq M$, la sottostruttura di $\mathcal{M}$ generata da D è la più piccola sottostruttura $\mathcal{N}$ di $\mathcal{M}$ tale che $D \subseteq N$. Per trovare $\mathcal{N}$ basta chiudere

$$D \cup \{c^{\mathcal{M}} : c \in \mathrm{Cost}\}$$

rispetto ad ogni $f^{\mathcal{M}}$ (questo fornisce il dominio N di $\mathcal{N}$) e poi definire $R^{\mathcal{N}}, f^{\mathcal{N}}, c^{\mathcal{N}}$ in accordo con le precedenti condizioni.

Questo corso comprende 4 dimostrazioni