Logica

Corso di Laurea in Informatica - Università degli studi di Verona

FEDERICO BRUTTI

Federico Brutti
federico.brutti@studenti.univr.it

Indice

Non ho ancora conosciuto una singola persona a cui sia piaciuta questa materia.

Introduzione

1.1 Che cos'è la logica matematica?

La **Logica Matematica** ha lo scopo di formalizzare concetti matematici in una lingua artificiale, per dare una certezza di significato al linguaggio naturale, il quale risulterebbe ambiguo. Similmente a quest'ultimo, anche i linguaggi della logica si compongono di grammatica, significato e regole, identificati rispettivamente in:

- **Sintassi**: Definisce la corretta scrittura delle formule logiche.
- **Semantica**: Definisce il significato delle formule logiche.
- **Sistemi Deduttivi**: Strumenti sintattici utili a manipolare formule e costruire dimostrazioni, dette derivazioni.

Abbiamo quindi un insieme di parole da assemblare (sintassi) che con diverse combinazioni possono creare sentenze dai diversi significati (semantica) ed il tutto segue un determinato insieme di regole (sistemi deduttivi). Tutto ciò consente di creare teoremi, anch'essi composti da più parti:

- **Enunciato**; Ciò che il teorema vuole esprimere.
- **Dimostrazione**; La prova formalizzata di quanto espresso.

Sarà nostro compito dimostrare gli enunciati in base alle richieste degli esercizi. Servirà specificare una procedura che permetta di ottenere solo risultati veri o falsi in linguaggio logico.

1.2 Connettivi e Quantificatori

Qui sono elencati tutti i connettivi e i quantificatori utilizzati nel corso, con lo scopo di familiarizzare con loro a prescindere da quando verranno o meno utilizzati.

- **Connettivi**
  - **Congiunzione** $\wedge$:
    Ritorna vero solo se tutti gli elementi sono veri.
  - **Disgiunzione** $\vee$:
    Ritorna vero se almeno un elemento è vero.
  - **Negazione** $\neg$:
    Rende falso il vero e viceversa.
  - **Implicazione** $\implies$:
    Corrisponde a "Se, allora", ritorna vero nei casi $0 \to 1$ oppure $1 \to 1$, mentre è falso se $1 \to 0$ oppure $0 \to 0$.

- **Doppia Implicazione** $\iff$:
  Corrisponde a "se e solo se, allora" e si rappresenta tramite due implicazioni: $(\phi \to \psi) \wedge (\psi \to \phi)$.
- **Bottom** $\perp$:
  Indica il valore di assurdo, 0.

• **Quantificatori**

- **Esiste** $\exists$:
  Indica l'esistenza di un elemento con una determinata proprietà.
- **Per ogni** $\forall$:
  indica che per ogni caso considerato, esiste un elemento con una data proprietà.

1.3 Strumenti di lavoro

Introduciamo in linguaggio naturale il funzionamento di tutti i nostri strumenti e strategie che utilizzeremo per la dimostrazione dei teoremi:

- **Sentenze**: Combinazioni dei singoli elementi dell'alfabeto. Equivalgono alle proposizioni in linguaggio naturale e possono assumere solamente due valori: vero o falso. Ne esistono di due tipi:
  - **Minimale**; Quando non si può scomporre ulteriormente.
  - **Composta**; Quando è composta da più sentenze minimali, ed è quindi scomponibile.
- **Induzione Strutturale**: Il principio di induzione è utilizzato per dimostrare la verità dell'enunciato a partire da un passo base semplice, per poi estendere l'ipotesi a ogni istanza successiva. Se l'ipotesi è provata, hai dimostrato l'enunciato.
- **Ricorsione Primitiva**: La ricorsione consiste nella formulazione di funzioni in termini di loro stesse all'interno della loro definizione. Le funzioni di una formula comprendono ogni caso possibile e riportano ad un passo base per terminare il processo.
- **Deduzione Naturale**: Processo di dimostrazione di una data formula a partire da delle ipotesi. Si tratta di un algoritmo basato puramente sulla correttezza sintattica.
- **Tabella di Verità**: Tabella che mostra ogni singolo caso presentabile per un determinato enunciato.
- **Valutazione di Verità**: Nella semantica, il processo di verifica della veridicità di una formula, esaminando il risultato di ogni singola formula presente nell'enunciato.
- **Sostituzione**: La sostituzione con una formulaψ tutte le occorrenze di un simbolo proposizionale o del primo ordine in un'altra formula ϕ.
- **Modello**: Definizione in termini logico-matematici di una data formula o funzione.
- **Contromodello**: Scrittura in termini logico-matematici che prova la fallacia di una data ipotesi iniziale.

Logica Proposizionale

Cominciamo lo studio effettivo della Logica Matematica attraverso la **Logica Proposizionale**. Fondamentalmente è una palestra da utilizzare come trampolino di lancio verso argomenti più complessi, come la Logica del Primo Ordine, che verrà vista in seguito.

Al termine del capitolo bisognerà saper valutare la veridicità di una formula e saper utilizzare la deduzione naturale per derivare sentenze.

2.1 Sintassi

In questo ambito il ragionamento si compone di entità linguistiche collegate dalla relazione "segue da". Queste si chiamano **Sentenze**. Come visto nell'introduzione possono essere minimali o composte, ma entrambe, se esprimono un pensiero completo si dicono **dichiarative**. Queste devono necessariamente avere un valore di vero o falso per essere considerate tali.

Esempio. Tipi di sentenze

- Sentenza minimale: ϕ

  Entità irriducibile.

- Sentenza composta: $\phi \vee \psi$

  Entità riducibile.

Ogni linguaggio logico è basato su un determinato **Alfabeto**, determinante i simboli utilizzabili in esso. Infatti, i linguaggi sono definiti come $\mathsf{L} \subseteq \mathsf{A}^*$, dove quest'ultimo insieme è quello delle **Stringhe**, ovvero tutte le sequenze di simboli generabili tramite l'alfabeto.

Le stringhe appartenenti ad L si dicono sintatticamente corrette. In particolare, definiamo ora l'alfabeto della logica proposizionale che consente la loro composizione:

Definizione 2.1. Alfabeto $\mathsf{L}^{\mathsf{PROP}}$
L'alfabeto della Logica Proposizionale è definito tramite tre unità sintattiche distinte:

- Insieme di simboli proposizionali $\mathsf{AT}^{\mathsf{PROP}}$, i cui elementi saranno indicati con lettere greche.
- Connettivi $\wedge$, $\vee$, $\rightarrow$, $\perp$.
- Simboli ausiliari "(" e ")". Non variano il senso della frase, ma aiutano a renderla più leggibile.

Abbiamo constatato che il linguaggio proposizionale è costituito dall'insieme delle stringhe sintatticamente corrette. Si indica con PROP e si definisce induttivamente come segue:

Definizione 2.2. Insieme delle proposizioni PROP

- $\perp \in \mathsf{PROP}$;
- Se p è un simbolo proposizionale, allora $p \in \mathsf{PROP}$;
- Se $\phi, \psi \in \mathsf{PROP}$, allora valgono le seguenti relazioni:
  $(\phi \wedge \psi) \in \mathsf{PROP}$, $(\phi \vee \psi) \in \mathsf{PROP}$, $(\phi \implies \psi) \in \mathsf{PROP}$, $(\neg\phi) \in \mathsf{PROP}$.
- Null'altro appartiene all'insieme PROP.

Per renderci la vita un pochino più semplice sono state ideate delle convenzioni sulla scrittura delle proposizioni; si possono infatti omettere alcune parentesi nelle formule e fissare un ordine di precedenza tra i connettivi:

- Si omettono le parentesi più esterne di una formula, così al posto di scrivere "$(\phi \wedge \psi)$", si scrive solo "$\phi \wedge \psi$".
- Il connettivo $\neg$ è il più forte e associa il primo simbolo presente a destra, perciò "$\neg\phi \vee \psi$" sarebbe uguale a "$(\neg\phi) \vee \psi$".
- Si assume che $\wedge$ e $\vee$ siano connettivi più forti rispetto a $\implies$ e $\iff$, quindi scrivere "$\phi_1 \wedge \psi_1 \implies \phi_2 \vee \psi_2$" è equivalente a intendere "$(\phi_1 \wedge \psi_1) \implies (\phi_2 \vee \psi_2)$".
- I connettivi $\implies, \wedge, \vee$ associano a destra, perciò scrivere "$\phi \implies \psi \implies \sigma$" equivale a dire: "$(\phi \implies (\psi \implies \sigma))$", mentre "$\phi \wedge \psi \vee \sigma$" sta per "$(\phi \wedge (\psi \vee \sigma))$".

Anche la logica matematica si basa sulla teoria degli insiemi; questo significa che valgono i teoremi inerenti, ma anche il funzionamento delle proprietà di un insieme. Diciamo infatti

Proposizione 2.3. Proprietà P su un insieme A
Se abbiamo un insieme P ed è sottoinsieme proprio o improprio di un altro A abbiamo che:

$$P \subseteq A \implies \text{proprietà } P \text{ su } A.$$

Di conseguenza, se abbiamo un elemento $a \in A$ possiamo dire che gode della proprietà se vale anche la relazione $a \in P$.

Finora abbiamo visto i soli elementi; ora di vedere cosa si può fare per dimostrare gli enunciati. Gli strumenti prendono la forma di **Induzione strutturale** e **Ricorsione primitiva**.

Teorema 2.4. Induzione strutturale su PROP
Considera una proprietà $P \subseteq \mathsf{PROP}$. Supponendo veri i punti:

- $\forall\phi \in \mathsf{AT}$, vale $P[\phi]$
  Se la proprietà vale per ogni formula atomica...
- $\forall\phi, \psi \in \mathsf{PROP}$, valgono $P[\phi], P[\psi]$
  E che se la proprietà vale per ogni formula in PROP...

Allora varranno le relazioni $\mathsf{P}[\phi \land \psi], \mathsf{P}[\phi \lor \psi], \mathsf{P}[\phi \implies \psi]$. Di conseguenza vale che $\forall \phi \in \mathsf{PROP}$, vale $\mathsf{P}[\phi]$.

Esempio. Costruzione induttiva della formula $\mathsf{P}[\phi \implies (\phi \lor \psi)]$
Fondamentalmente devi scomporre l'intera formula. Parti dalle formule atomiche

$$\mathsf{P}[\phi], \mathsf{P}[\psi]$$

Entrambe valgono, quindi puoi prendere una o l'altra. Puoi inoltre introdurre l'implicazione.

$$\mathsf{P}[\phi] \implies \mathsf{P}[\phi \lor \psi] := \mathsf{P}[\phi \implies (\phi \lor \psi)$$

E hai finito.

Parliamo ora di ricorsione. Premetto che tutte le volte in cui si usa un linguaggio definito induttivamente, è possibile descriverne gli elementi anche ricorsivamente. Sempre. Questo algoritmo genera funzioni in termini di sé stesse all'interno della propria definizione. Risulta più semplice con l'esempio del fattoriale:

Esempio. Funzione fattoriale

$$fatt = \begin{cases} 0! = 1 \\ n! = n \times (n-1)! \end{cases}$$

Come si può vedere, la funzione comprende ogni singolo caso che si può presentare. Se alla funzione è dato il valore 0 sei nel caso base, mentre in qualunque altra istanza sostituisci il numero dato alla n e poi svolgi i calcoli.

Terminiamo con due ultimi concetti sintattici definiti ricorsivamente:

Definizione 2.5. Rango di una formula
Definiamo $r : \mathsf{PROP} \to \mathbb{N}$ ricevente in input una formula, il valore del suo rango. Si tratta di una quantità statistica utile al fine di vedere una definizione per ricorsione.

- $r[\phi] = 0$, se $\phi \in AT$
- $r[(\neg\phi)] = 1 + r[\phi]$
- $r[(\phi \circ \psi)] = 1 + max\{r[\phi], r[\psi]\}$, con $\circ \in \{\lor, \land, \implies\}$

Definizione 2.6. Sottoformula
Si definisce sottoformula $\mathsf{sub} : \mathsf{PROP} \to 2^{\mathsf{PROP}}$ la funzione che prende in input una formula proposizionale e ne restituisce l'insieme delle parti. Compone l'insieme di tutte le sottoformule presenti in una certa formula data.

- $\mathsf{sub}[\phi] = \{\phi\}$, se $\phi \in \mathsf{AT}$
- $\mathsf{sub}[(\neg\phi)] = \{(\neg\phi)\} \cup \mathsf{sub}[\phi]$
- $\mathsf{sub}[(\phi \circ \psi)] = (\phi \circ \psi) \cup \mathsf{sub}[\phi] \cup \mathsf{sub}[\psi]$, con $\circ \in \{\wedge, \vee, \implies\}$

2.2 Semantica

Finora abbiamo studiato il valore sintattico delle frasi, ma non dimentichiamo che, come nella lingua italiana, una composizione di frasi può avere un significato; nel linguaggio logico è necessario analizzare gli enunciati per determinarne eventualmente verità, col valore di 1, o falsità, col valore di 0. Come fare?

Definizione 2.7. Valutazione di verità
Sia la funzione $V : \mathsf{PROP} \to \{0, 1\}$, ovvero $V \in \mathsf{PROP}$ che può ritornare uno dei due valori fra $0, 1$. Questa è tale che:

- $V[\bot] = 0$
- $V[\neg\psi] = 1 \iff V[\psi] = 0$
- $V[\psi \wedge \phi] = 1 \iff V[\psi] = 1 \wedge V[\phi] = 1$
- $V[\psi \vee \phi] = 1 \iff V[\psi] = 1 \vee V[\phi] = 1$
- $V[\psi \to \phi] = 1 \iff V[\psi] = 0 \vee V[\phi] = 1$
- $V[\psi \iff \phi] = 1 \iff V[\psi] = V[\phi]$

Si ricorda più facilmente pensando alle tavole di verità delle varie funzioni. Difatti, è la loro definizione ricorsiva.

Ora, considera le variabili che abbiamo usato finora; queste possono indicare composizioni di formule al loro interno, quindi non essere atomiche, si definiscono infatti **metavariabili**, e per conoscere il loro valore semantico è necessario valutare i loro simboli proposizionali. Ciò avviene grazie alla **Valutazione atomica**, la base per definire tutte le valutazioni di verità.

Il concetto è definito in modo astratto, ma fondamentalmente devi scomporre la formula fin quando ogni suo elemento diventa atomico. Dopo aver valutato i singoli valori, ricava il risultato in base ai connettivi usati.

Definizione 2.8. Valutazione atomica
Sia la seguente funzione:

$$V : \mathsf{AT} \to \{1, 0\} \text{ tale che } \nu[\bot] = 0.$$

Data una valutazione atomica ν, esiste ed è unica una valutazione:

$$[\cdot]_\nu : \mathsf{PROP} \to \{1, 0\}, \text{ Tale che } [\alpha]_\nu = \nu[\alpha] \text{ per } \alpha \in \mathsf{AT}.$$

Nel caso in cui nell'esercizio non venga richiesto se cercare verità o falsità, si valuta ogni singolo caso. Vediamo ora un esempio:

Esempio. Effettuare valutazione di verità sulla formula: $(\phi \wedge \psi) \implies (\neg\psi \vee \phi)$.
Partiamo da un ragionamento semplice: avendo un'implicazione, le uniche istanze che possono riportare un valore vero, data la formula $P \implies Q$, sono $0 \to 1$, $1 \to 1$, $0 \to 0$.

Muoviamoci ora per determinare i valori delle singole formule atomiche, una metavariabile alla volta, in base alle istanze discusse:

- $P = [(\phi \wedge \psi)]_v = 0 \iff [\phi]_v = 0$ OR $[\psi]_v = 0$.
- $Q = [(\neg\psi \vee \phi)]_v = 1 \iff [\neg\psi]_v = 1$ OR $[\phi]_v = 1 \iff [\psi]_v = 0$ OR $[\phi]_v = 1$.

Nella formula P abbiamo ricavato che almeno una delle due variabili deve avere un valore di falsità. Guardando infatti la formula Q, notiamo che $[\psi]_v = 0$. L'ipotesi è confermata e non andando incontro a contraddizioni, abbiamo dimostrato che la formula è **vera**.

Si possono inoltre presentare alcuni casi notevoli davanti alla valutazione del significato di un enunciato, come:

Definizione 2.9. Concordanza e Tautologia

- Dato un insieme finito di proposizioni P, se due valutazioni differenti risultano uguali, si dicono concordanti su P.
- Se una formula proposizionale è vera in ogni caso, si dice **Tautologia** e si indica con $[\alpha]_v = 1$.

Negli esercizi verranno utilizzate le seguenti scritture, con $\alpha \in PROP$ generica:

- $\vDash \alpha$: Tautologia.
- $\vdash \alpha$: Teorema.

Per dimostrare una tautologia è necessario controllare che sia vera per ogni singolo caso; se in un'istanza si rivela falsa, non è tale.

Teorema 2.10. Tautologie Notevoli

- Leggi di De Morgan:

$$\vDash \neg(\phi \wedge \psi) \iff (\neg\phi \vee \neg\psi), \vDash \neg(\phi \vee \psi) \iff (\neg\phi \wedge \neg\psi).$$

- Negazione Involutiva:

$$\vDash \phi \iff \neg\neg\phi.$$

- Leggi Commutative:

$$\models (\phi \wedge \psi) \iff (\psi \wedge \phi), (\phi \vee \psi) \iff (\psi \vee \phi).$$

- Leggi Distributive:

$$\models \phi \wedge (\psi \vee \gamma) \iff ((\phi \wedge \psi) \vee (\phi \wedge \gamma)), \phi \vee (\psi \wedge \gamma) \iff ((\phi \vee \psi) \wedge (\phi \vee \gamma)).$$

- Legge Associativa:

$$\models \phi \wedge (\psi \wedge \gamma) \iff (\phi \wedge \psi) \wedge \gamma, \phi \vee (\psi \vee \gamma) \iff (\phi \vee \psi) \vee \gamma.$$

L'algoritmo di valutazione della verità può essere esteso anche ad un insieme determinato di proposizioni, che chiameremo Γ. Definiamo quindi:

Definizione 2.11. Soddisfacibilità di una formula e conseguenza logica
La valutazione di verità si estende ad un insieme di proposizioni arbitrario Γ tale che:

$$[\Gamma]_v = 1 \iff \forall \alpha \in \Gamma \mid [\alpha]_v = 1.$$

Attenzione, se $[\Gamma]_v \neq 1$, significa che c'è *almeno una* valutazione dove risulta falsa, ma non che lo sono tutte. Se si afferma invece che $[\Gamma]_v = 0$, tutte le valutazioni saranno false.

Da questo insieme di proposizioni è possibile anche trarre delle conclusioni, dette **conseguenze logiche**. Si dice che una proposizione ϕ segue logicamente da un insieme Γ se la verità delle proposizioni in esso contenute implica la verità di ϕ.

Dato infatti l'insieme Γ di formule, si dice ψ conseguenza logica delle formule di Γ e si scrive:

$$\Gamma \models \psi, \text{ ovvero in questo caso: } [\Gamma]_v = 1 \implies [\psi]_v = 1.$$

Inoltre, se un insieme vuoto implica la verità di una formula, abbiamo dimostrato una tautologia.

Diremo infine che una formula ψ è detta **soddisfacibile** se esiste una valutazione v per cui risulti vera, altrimenti è **insoddisfacibile**. Il concetto si può estendere agli insiemi di proposizioni; Γ si dice soddisfacibile quando una sua formula ψ è sempre vera data una certa valutazione, altrimenti è insoddisfacibile. Formalmente:

Esempio. Soddisfacibilità di un insieme Γ

- Se $\{p\} \in \Gamma$, $[p]_v = 1$ e quindi l'insieme è soddisfacibile.
- Se $\{p, \neg p\} \in \Gamma$, non tutte le valutazioni di *p* sono vere in quanto contraddittorie e quindi l'insieme non è soddisfacibile.

Le proprietà della conseguenza logica saranno riprese in seguito nella sezione dei sistemi deduttivi, in quanto saranno usate per la deduzione naturale. Parliamo ora invece di **Sostituzione proposizionale**, una tecnica che consente, in una formula ϕ, di sostituire tutte le occorrenze di una formula p_1 con un'altra ψ.

Definizione 2.12. Sostituzione proposizionale

$$\text{Sia la formula: } \phi = ((p_1 \to (p_5 \vee p_1)) \wedge p_3).$$

Qui è possibile sostituire tutti i simboli p_1 con ψ. Il passaggio intermedio si scrive $\phi[\psi/p_1]$, ovvero che ψ sostituirà il valore p_1. Risulterà:

$$\phi = ((\psi \to (p_5 \vee \psi)) \wedge p_3).$$

Se un termine nella formula presa in esame non compare affatto, è ovvio che la sostituzione non avrà alcuno scopo o effetto.

Seguono ora le proprietà dell'algoritmo appena discusso:

Proposizione 2.13. Sia la funzione $[\psi/p] : \mathsf{PROP} \to \mathsf{PROP}$ tale che:

- $\phi[\psi/p] = \bot$ se $\phi = \bot$.
- $\phi[\psi/p] = \phi$ se $\phi \in \mathsf{AT} \wedge \phi \neq p$.
- $\phi[\psi/p] = \psi$ se $\phi = p$.
- $\neg(\phi)[\psi/p] = \neg(\phi[\psi/p])$.
- $(\phi_1 \circ \phi_2)[\psi/p] = (\phi_1[\psi/p] \circ \phi_2[\psi/p])$, dove $\circ = \{\wedge, \vee, \implies\}$.

E ora sblocchiamo il vero potenziale della sostituzione. Ragiona: se due formule risultano uguali, avranno egual valore nelle valutazioni. Se sono tali, sono intercambiabili, o anche meglio, scriverle allo stesso modo. Ciò ci consente di raggruppare più formule **equivalenti** sotto uno stesso termine, rendendo l'enunciato nettamente più semplice, infatti:

Proposizione 2.14. Sostituzione di formule equivalenti
Siano due formule $\phi_1\,\phi_2$ e $p \in \mathsf{AT}$. Se le prime due formule sono equivalenti sintatticamente, ovvero $\vDash \phi_1 \longleftrightarrow \phi_2$, prendendo una formula generica ψ possiamo scrivere:

$$\psi[\phi_1/p] \longleftrightarrow \psi[\phi_2/p].$$

La relazione di **equivalenza sintattica** $\longleftrightarrow$ è data dalle seguenti proprietà:

$$\text{Siano } \vDash \phi_1 \longleftrightarrow \phi_1', \vDash \phi_2 \longleftrightarrow \phi_2'$$

Varranno le seguenti relazioni:

- $\vDash \phi_1 \implies \phi_2 \longleftrightarrow \phi_1' \implies \phi_2$.
- $\vDash \phi_1 \wedge \phi_2 \longleftrightarrow \phi_1' \wedge \phi_2'$.
- $\vDash \phi_1 \vee \phi_2 \longleftrightarrow \phi_1' \vee \phi_2'$.

Formalizziamo infine il concetto di **Relazione di equivalenza**. Molto semplicemente, i simboli $\longleftrightarrow$, $\iff$ Rappresentano rispettivamente equivalenza sintattica e semantica; sono una scrittura compatta del seguente concetto:

$$\phi \iff \psi = (\phi \to \psi) \land (\psi \to \phi).$$

Restituisce il valore di vero solamente quando entrambe le formule sono vere.[1]
Una relazione di equivalenza R si scrive $\approx$ ed è tale se vengono rispettati questi tre criteri:

1. **Riflessività**; $\forall a \in A \mid aRa$.
2. **Transitività**; $\forall a, b, c \in A \mid (aRb \land bRc) \to aRc$.
3. **Simmetria**; $\forall a, b \in A \mid aRb \to bRa$.

Teorema 2.15. Relazione di equivalenza

Date due formule $\phi, \psi \in \mathsf{PROP}$, se $\vDash \phi \leftrightarrow \psi$, si scriverà $\phi \approx \psi$.

2.3 Sistemi Deduttivi

Prima di iniziare a lavorare, è necessario introdurre il concetto di **Sistema Deduttivo**. Si tratta di un costrutto che fornisce una struttura formale di regole inferenziali, le quali permettono di derivare una conclusione a partire da alcune premesse.

Nell'enunciare un teorema, è necessario fornire anche la sua dimostrazione, poiché, senza di essa, definiamo il primo **congettura**. La dimostrazione di un teorema è definita come un algoritmo: sequenza di passi non ambigui mirati ad arrivare alla tesi partendo dalle ipotesi. I passi sono tutto lo svolgimento che effettueremo per dimostrare l'enunciato e si chiama **deduzione**.

Definizione 2.16. Deduzione Naturale
Metodo di ragionamento puramente sintattico. Si effettua stabilendo:

- **Assunzioni**; Ragionamenti temporanei.
- **Derivazioni**; Dalle formule è possibile effettuare derivazioni attraverso determinate leggi. Ogni passaggio si dice **Passo di derivazione**.
- **Scaricamento**; La chiusura delle assunzioni fatte all'inizio.

Ogni derivazione è rappresentata tramite **alberi**, dove le **foglie** sono le ipotesi e la **radice** l'enunciato. Una dimostrazione si dice completa quando ogni foglia è chiusa, ovvero quando si è scaricata ogni ipotesi; si indica con $\mathsf{Hp}[\mathsf{D}] = \emptyset$.
La scrittura significa che l'insieme delle ipotesi della derivazione D è vuoto. Possono inoltre esistere più derivazioni per un singolo teorema.

Passiamo ora alle **Regole deduttive**. Si tratta di algoritmi utili per la derivazione delle formule nella deduzione naturale. Si dividono in due tipi:

[1] Ciò implica anche che se le formule sono equivalenti e la valutazione di una delle due è vera, lo sarà per forza anche l'altra.

- **Introduzione**; Premesse collegate introducendo un connettivo logico.
- **Eliminazione**; Dalle premesse è possibile rimuovere un connettivo logico.

Negli esercizi, le regole deduttive consentono di effettuare i passi di derivazione e sono rappresentati in forma di frazione. Sopra stanno le premesse, sotto la conclusione e a fianco la regola utilizzata. Le formule all'interno di parentesi quadre sono poi ipotesi assunte/scaricate da una determinata regola di derivazione, la quale dovrà essere appropriatamente numerata. Vediamo come fare insieme alla rappresentazione delle regole a nostra disposizione:

- **Regole dell'implicazione** ($\rightarrow$)
  - **Introduzione dell'implicazione**
    Se dall'ipotesi α segue β, con una certa derivazione D, allora si deriva $\alpha \rightarrow \beta$ tramite l'introduzione dell'implicazione. Questa regola assume/scarica come ipotesi l'implicante dell'implicazione.

    $$\frac{[\beta]^1}{\alpha \rightarrow \beta} \rightarrow I^1$$
  - **Eliminazione dell'implicazione**
    Chiamato anche *modus ponens*, se da due derivazioni D_1, D_2 otteniamo α e $\alpha \rightarrow \beta$, possiamo concludere β eliminando il connettivo.

    $$\frac{\alpha \qquad \alpha \rightarrow \beta}{\beta} \rightarrow E^1$$
  - **Indebolimento**
- **Regole della congiunzione** ($\wedge$)
  - **Introduzione dell'AND**
  - **Eliminazione dell'AND**
- **Regole del bottom** ($\perp$)
  - **Riduzione ad assurdo (RAA)**
  - **Eliminazione del bottom**
- **Regole della disgiunzione** ($\vee$)
  - **Introduzione dell'OR**
  - **Eliminazione dell'OR**
- **Regole della doppia implicazione** ($\leftrightarrow$)
  - **Introduzione della doppia implicazione**
  - **Eliminazione della doppia implicazione**
- **Assiomi**

2.4 Correttezza e Completezza

2.5 Esercizi

Logica del Primo Ordine

La **Logica del Primo Ordine**, o dei Predicati è un'espansione della Logica Proposizionale. La novità principale è la presenza di due nuovi simboli detti **Quantificatori**: *Per ogni* $\forall$ ed *Esiste* $\exists$, con i quali sarà possibile descrivere strutture matematiche. La loro particolarità è che legano più di ogni altro connettivo.

3.1 Sintassi

3.2 Sistemi Deduttivi

formalizzazione, contromodello, struttura di Peano

3.3 Semantica

3.4 Esercizi

3.5 Appunti

3.5.1 Strutture e Tipi di Similarità

3.5.2 Derivazione Semantica

3.5.3 Deduzione Naturale

3.5.4 Teoremi di Correttezza e Completezza