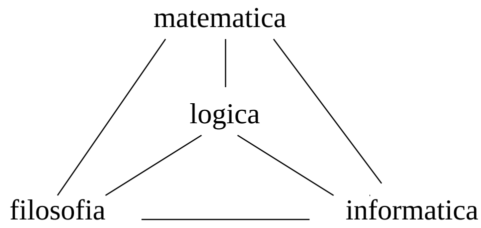
4 La Logica come base di ogni scienza

La **Logica** è alla **base di ogni scienza (o teoria)** in quanto è **fondamento di ogni scienza** non tanto per i contenuti specifici ma per la loro articolazione **deduttiva**.



Infatti la logica si occupa di riconoscere la verità di un enunciato non tanto in quanto corrisponde ad uno stato del mondo (come avviene per le scienze) quanto di stabilire le **condizioni di verità** di un enunciato a partire da altri basandoci solo sulla sua **forma logica** espressa in un *LINGUAGGIO FORMALE*.

Prima di introdurre il linguaggio formale cerchiamo di capire cosa sia una forma logica di un enunciato intuitivamente.

5 Alla ricerca della forma logica

Intuitivamente la forma logica di una proposizione è la struttura astratta dei legami logici delle proposizioni semplici che la compongono.

Ad esempio la proposizione dell'asserzione

> Ammesso che
> "**se piove, i canali interni vengono chiusi**",
> allora, è vero che
> "**se i canali interni non vengono chiusi, allora non piove**"

è complessa ed è composta dalle seguenti proposizioni semplici: "**piove**" e "**i canali interni vengono chiusi**" legate logicamente tramite "Amesso che", e "allora è vero che".

La proposizione della seguente asserzione ha la stessa forma logica:

> È vero che
> "**se il tuo vicino di banco non è Napoleone**
> **ne segue che la radice quadrata non canta alla Scala di Milano**"
> ammesso che
> "**se la radice quadrata canta alla Scala di Milano**
> **allora il tuo vicino di banco è Napoleone**".

che ha pure la stessa forma e contenuto di **esercizio 3. del test**.

Mentre l'asserzione

> "È vero che **c'è silenzio**
> se **tutti dormono**
> ed se è vero che
> **se tutti dormono c'è silenzio.**"

ha la stessa forma di

> "Ammesso che **Tizio ama Caio**
> e che **se Tizio ama Caio allora Caio ama Tizio.**
> ne segue che **Caio ama Tizio**".

Ora introduciamo un modo più semplice per fare asserzioni complesse come quelle sopra.

Si noti che nelle asserzioni sopra, c'è sempre una *conclusione* (per esempio **Caio ama Tizio** nell'asserzione immediatamente sopra) che segue da delle proposizioni assunte come vere, dette *premesse*, (nell'asserzione immediatamente sopra le premesse sono **Tizio ama Caio** e **se Tizio ama Caio allora Caio ama Tizio**).

Ora indichiamo tali asserzioni complesse mettendo le premesse in lista sopra una linea che separa la conclusione come segue: scriviamo

$$\frac{\begin{array}{l}\textbf{Tizio ama Caio.}\\ \textbf{se Tizio ama Caio allora Caio ama Tizio.}\end{array}}{\textbf{Caio ama Tizio.}}$$

come forma concisa per

"Ammesso che **Tizio ama Caio** e che **se Tizio ama Caio allora Caio ama Tizio.** ne segue che **Caio ama Tizio**".

Altro esempio: scriviamo

$$\frac{\begin{array}{l}\textbf{Tutti dormono.}\\ \textbf{Se tutti dormono c'è silenzio.}\end{array}}{\textbf{C'è silenzio.}}$$

come forma concisa per

" È vero che "**c'è silenzio**" se **tutti dormono** ed se è vero che **se tutti dormono c'è silenzio.**"

5.1 Necessità di un linguaggio formale

Per descrivere la **forma logica** di una frase si definisce un **linguaggio formale** (o **linguaggio simbolico**).

Ogni *linguaggio di programmazione* è un *esempio* di *linguaggio formale*.

Prima di introdurre il concetto di linguaggio formale torniamo sulla distinzione tra livelli di riferimento (per approfondimento il lettore legga il capitolo 1 del libro di Sambin).

5.1.1 Livelli di riferimento in un programma

Nel programma

$$\begin{array}{l} y = 1;\\ z = 0;\\ \texttt{while}\quad (z \neq x)\quad \{\\ \qquad\qquad z \ = \ z+1\\ \qquad\qquad y \ = \ y * z;\\ \quad \} \end{array}$$

quanti **livelli** di **astrazione** o **riferimento** ci sono?

1. **codice del programma**, $\Rightarrow$ livello del "**linguaggio**" cioè **sintassi**;

2. **commento/verifica** di ciò che fa, $\Rightarrow$ livello del **metalinguaggio** cioè **SEMANTICA**

5.2 Livelli di riferimento nel corso

Nel nostro corso parleremo di almeno 2 livelli di riferimento in relazione ai linguaggi formali:

1. livello del **linguaggio formale — sintassi**
2. livello del **metalinguaggio/nostro linguaggio naturale — semantica**

Il livello del **linguaggio formale** è costituito da simboli ed espressioni del linguaggio che possiamo associare in modo specifico ad una MACCHINA o ROBOT.

Invece il livello del **metalinguaggio** è dato dal significato dei simboli ed espressioni del precedente livello che è assegnato da NOI in modo specifico.

Ricordiamo che dobbiamo operare una netta distinzione tra tali livelli di riferimento per non incorrere in paradossi.

5.3 Spiegazione del carattere ASTRATTO della LOGICA

Ora possiamo capire meglio la citazione di Russell

In **logica**, come in **matematica**
non si sa **di cosa si parla**
nè se **ciò di cui si parla** sia **vero**.

ricordando che lo scopo della logica è di introdurre un **linguaggio simbolico** per studiare la **FORMA** degli enunciati **SENZA RIFERIMENTO** al contenuto semantico specifico dei loro componenti atomici.

Ad esempio l'argomentazione

$$\frac{\textbf{Nessun falipo è goloso e Giove è un falipo.}}{\textbf{Giove non è goloso.}}$$

è valida logicamente, come l'esercizio 3. del test anche se la parola "falipo" non compare nel vocabolario italiano, nè sappiamo se abbia senso attribuirgli l'aggettivo "goloso".

6 Linguaggio formale proposizionale

Costituenti delle nostre asserzioni sono le proposizioni, ove con proposizione si intende un enunciato in un determinato linguaggio, non solo dotato di senso ma anche di valore di verità (per approfondimento si rimanda al capitolo 1. del libro di Sambin "**Per Istruire Un Robot**: **(ovvero, come costruirsi una logica)**". La logica formale studia le forme logiche delle proposizioni e la loro validità.

Ora introduciamo un linguaggio formale contenente segni per denotare le **PROPOSIZIONI** che si distinguono in atomiche e composte.

A tal fine usiamo le lettere dell'alfabeto

$$\mathbf{A}, \mathbf{B}, \mathbf{C} \ldots, \mathbf{Z}$$

come NOMI per indicare proposizioni atomiche qualsiasi in modo **ASTRATTO**. Nel gergo formale $\mathbf{A}, \mathbf{B}, \mathbf{C} \ldots, \mathbf{Z}$ si dicono **VARIABILI PROPOSIZIONALI**.

A partire dalle **variabili proposizionali**

$$\mathbf{A}, \mathbf{B}, \mathbf{C} \ldots, \mathbf{Z}$$

costruiamo proposizioni composte usando i segni di
connettivo unario della **negazione**

$$\neg$$

connettivo binario dell' **implicazione**

$$\rightarrow$$

connettivo binario della **congiunzione**

$$\&$$

connettivo binario della **disgiunzione**

$$\vee$$

6.1 Grammatica delle proposizioni formali

Una proposizione formale

$$\mathtt{pr}$$

(che è una META-variabile per indicare una proposizione formale generica)

è una stringa di simboli ottenuti in tal modo:
pr $\equiv$ **A** oppure **pr** $\equiv$ **B** oppure una qualsiasi variabile proposizionale, che noi abbiamo fissato essere una lettera dell'alfabeto;
oppure **pr** coincide con una delle seguenti proposizioni ottenute da altre due generiche proposizioni $\mathbf{pr_1}$ e $\mathbf{pr_2}$ come segue:

$\mathbf{pr_1}\&\mathbf{pr_2}$ che sta per $\mathbf{pr_1}$ **e** $\mathbf{pr_2}$
$\mathbf{pr_1}\vee\mathbf{pr_2}$ che sta per $\mathbf{pr_1}$ **o** $\mathbf{pr_2}$
$\mathbf{pr_1}\rightarrow\mathbf{pr_2}$ che sta per **se** $\mathbf{pr_1}$ **allora** $\mathbf{pr_2}$
ovvero $\mathbf{pr_1}$ **implica** $\mathbf{pr_2}$
$\neg\mathbf{pr_1}$ che sta per **NON si dà il caso che** $\mathbf{pr_1}$

6.2 Esempi di proposizioni simboliche

1. La proposizione
   "**Oggi è venerdì e domani è sabato**"
   ha la forma logica di congiunzione di due proposizioni

   $$\mathbf{V}\&\mathbf{S}$$

   ove
   V= "**Oggi è venerdì**"
   S= "**domani è sabato**"

2. "Oggi è venerdì e domani è sabato, mentre dopodomani è domenica" si può formalizzare così

   $$(V\&S)\&D$$

   ove
   V="Oggi è venerdì"

S=“domani è sabato”
D=“dopodomani è domenica”

(si noti che “mentre” ha lo stesso significato di una &)

3. “Solo se piove prendo l’ombrello”

   si può formalizzare così
   $$O \rightarrow P$$
   ove
   O = prendo l’ombrello e P = piove.

   (Si noti che il fatto che “piova” è la condizione NECESSARIA affinchè io porti l’ombrello...).

4. “Il programma fattoriale termina sull’input 5 perchè ad un certo punto la condizione del `while` diventa falsa.”

   si può formalizzare così
   $$F\&T$$
   ove
   F=“ad un certo punto la condizione del `while` diventa falsa”
   T=“Il programma fattoriale termina sull’input 5”

5. “Solo se ad un certo punto la condizione del `while` diventa falsa allora il programma fattoriale termina sull’input 5.”

   si può formalizzare così
   $$T \rightarrow F$$

6. “Se ad un certo punto la condizione del `while` diventa falsa allora il programma fattoriale termina sull’input 5.”

   si può formalizzare così
   $$F \rightarrow T$$

7. “Se e solo se ad un certo punto la condizione del `while` diventa falsa allora il programma fattoriale termina sull’input 5.”

   si può formalizzare così
   $$(F \rightarrow T)\ \&\ (T \rightarrow F)$$

8. “Ad un certo punto la condizione del `while` diventa falsa e quindi il programma fattoriale termina sull’input 5.”

   si può formalizzare così
   $$F\&T$$
   (si noti che la frase sopra esprime non solo che vale $F\&T$ ma anche c‘è un legame causale tra T ed F, ovvero che F implica T, ovvero che vale $F \rightarrow T$ oltrechè F da cui segue T).

   ove da 4) in poi le lettere F e T stanno ad indicare le proposizioni come in 3).

6.3 Cosa traducono & ed $\rightarrow$

Si noti che la congiunzione $\mathbf{pr_1} \& \mathbf{pr_2}$ traduce legami tra $\mathbf{pr_1}$ e $\mathbf{pr_2}$ del tipo

$\mathbf{pr_1}$ *e* $\mathbf{pr_2}$	$\mathbf{pr_1}$ *perchè* $\mathbf{pr_2}$	$\mathbf{pr_1}$ *mentre* $\mathbf{pr_2}$
$\mathbf{pr_1}$ *però* $\mathbf{pr_2}$	$\mathbf{pr_1}$ *quindi* $\mathbf{pr_2}$	$\mathbf{pr_1}$ *ma* $\mathbf{pr_2}$

mentre l' *implicazione* $\mathbf{pr_1} \rightarrow \mathbf{pr_2}$ traduce legami del tipo

se $\mathbf{pr_1}$ *allora* $\mathbf{pr_2}$	$\mathbf{pr_1}$ *solo se* $\mathbf{pr_2}$
$\mathbf{pr_2}$ *se* $\mathbf{pr_1}$	*solo se* $\mathbf{pr_2}$ *vale* $\mathbf{pr_1}$

6.4 Formalizzazione di enunciati con premesse e conclusioni

Diamo ora la formalizzazione logica di enunciati più complessi come quelli in sezione 5 ove una conclusione segue da una o più premesse.

Ad esempio l'asserzione

"È vero che **se il treno è in ritardo i viaggiatori non sono contenti** se si assume che **se i viaggiatori son contenti allora il treno non è in ritardo**".

si può formalizzare come UNICO enunciato formale in

$$(V \rightarrow \neg R) \rightarrow (R \rightarrow \neg V)$$

ove
V="**i viaggiatori sono contenti**"
R="**il treno è in ritardo**"

ma grazie alla convenzione nella sezione 5 possiamo anche più semplicemente formalizzarlo in tal modo

$$\frac{V \rightarrow \neg R}{R \rightarrow \neg V}$$

Diamo di seguito la formalizzazione di altre asserzioni:

1. L'asserzione

   "È vero che **non si dà il caso che non ci sia silenzio** se **tutti dormono** e se è vero che **se tutti dormono c'è silenzio**."

   si può formalizzare in

   $$D \& (D \rightarrow S) \ \rightarrow \ \neg\neg S$$

   e secondo la convenzione in sezione 5 in

   $$\frac{\begin{array}{l} D \\ D \rightarrow S \end{array}}{\neg\neg S}$$

   ove

D= "**tutti dormono**"
S="**c'è silenzio**"

2. L'asserzione

   "Ammesso che **Tizio ama Caio** e che **se Tizio ama Caio allora Caio ama Tizio** ne segue che **Caio ama Tizio**".

   si può formalizzare in

   $$T\&(T \rightarrow C) \ \rightarrow \ C$$

   e secondo la convenzione in sezione 5 in

   $$\begin{array}{l} T \\ T \rightarrow C \\ \hline C \end{array}$$

   ove
   T= "**Tizio ama Caio**"
   C="**Caio ama Tizio**"

3. L'asserzione

   "Ammesso che **se piove, i canali interni vengono chiusi**, allora, è vero che **se i canali interni non vengono chiusi, allora non piove**".

   si può formalizzare in

   $$(P \rightarrow C) \ \rightarrow \ (\neg C \rightarrow \neg P)$$

   e secondo la convenzione in sezione 5 in

   $$\begin{array}{l} P \rightarrow C \\ \hline \neg C \rightarrow \neg P \end{array}$$

   ove
   P= "**piove**"
   C="**i canali interni vengono chiusi**"

4. L'asserzione

   "È vero che **c'è silenzio** se **tutti dormono** e se è vero che **se tutti dormono c'è silenzio**."

   si può formalizzare in

   $$D\&(D \rightarrow S) \ \rightarrow \ S$$

   e secondo la convenzione in sezione 5 in

   $$\begin{array}{l} D \\ D \rightarrow S \\ \hline S \end{array}$$

   ove

D= "**tutti dormono**"
S="**c'è silenzio**"

5. l'asserzione

   "È vero che **se il tuo vicino di banco non è Napoleone ne segue che la radice quadrata non canta alla Scala di Milano** se si suppone che **se la radice quadrata canta alla Scala di Milano allora il tuo vicino di banco è Napoleone**"

   si può formalizzare in

   $$(C \to N) \;\to\; (\neg N \to \neg C)$$

   e secondo la convenzione in sezione 5 in

   $$\frac{C \to N}{\neg N \to \neg C}$$

   ove
   N= "**il tuo vicino di banco è Napoleone**"
   C="**la radice quadrata canta alla Scala di Milano**"

Come si vede sopra le proposizioni in 3) e 5) hanno la stessa forma logica (a meno di variabili proposizionali), e così pure 2) e 4).

6.5 ATTENZIONE: come mettere le parentesi

Nello scrivere le proposizioni simboliche $\neg$ si lega alla formula più vicina più di ogni altro connettivo, seguito a pari merito da $\vee, \&$, che a loro volta sono legate alle formule più di $\to$.
Ovvero

$$\neg \quad \textbf{lega più di} \quad \vee, \& \quad \textbf{lega più di} \quad \to$$

Esempi:

"(**negazione** di **A**) **o B**"

si scrive

$$\neg\mathbf{A}\vee\mathbf{B}$$

"**negazione** di **(A o B)**"

si scrive

$$\neg(\mathbf{A}\vee\mathbf{B})$$

"la **(** **negazione** di A **) implica (** B **o** C **)**"

si scrive

$$\neg\mathbf{A}\to\mathbf{B}\vee\mathbf{C}$$

"la **negazione** di ((A **implica** B) **o** C) "

si scrive

$$\neg(\ (\mathbf{A}\rightarrow\mathbf{B})\vee\mathbf{C}\ )$$

6.6 Alla ricerca della verità

Ricordando che la **Logica** si occupa di studiare la **verità** di un'argomentazione o proposizione **SOLTANTO** in base alla sua **forma logica** definiamo allora la **verità** di una **proposizione**.

Precisiamo inoltre che propriamente una **proposizione** è tale se oltre ad essere una successione di segni dotati di significato abbiamo anche un *criterio* per *stabilire se la proposizione è* **vera** *o* **falsa**.

Ricordiamo ad esempio che

"**Questa proposizione è falsa**"

NON è una proposizione propriamente, in quanto non può essere nè vera nè falsa come argomentato in sezione 3.4.

Altri esempi e controesempi di proposizioni sono i seguenti:

"**In biblioteca al piano interrato c'è l'ultimo numero di Topolino**"
è una proposizione, perchè posso verificare la sua verità andando in biblioteca.

"**La radice quadrata canta alla Scala di Milano.**"
non è una proposizione, non ha senso.

6.6.1 In logica solo giudizi assertivi

In logica studiamo soltanto le **PROPOSIZIONI ASSERITE** come **VERE**. Una proposizione asserita come vera si dice **giudizio assertivo**.

Ad esempio:
Affermo che "**il numero 3 è dispari**".
oppure semplicemente "**il numero 3 è dispari**"

6.6.2 Cosa è un giudizio in generale?

Un **giudizio** è l'atto di dichiarare una proposizione e vi sono DIVERSI tipi di giudizio:

assertivo= la proposizione è asserita vera
ad es. "**Mario studia.**"

interrogativo= s'interroga sulla verità della proposizione
ad es."**Mario studia?**"

direttivo= la proposizione esprime un'ordine da eseguire
ad es. "**Studia, Mario!**"

esclamativo= la proposizione esprime un'esclamazione.
ad es. "**Mario studia!**"

6.7 Verità classica di una proposizione

Per stabilire quando una proposizione formale è vera ci serviamo delle **tabelle di verità**. A tal scopo ad ogni proposizione formale **pr** costruita tramite COMPOSIZIONE di connettivi logici

$$\mathbf{pr} \equiv \mathbf{comp}(\mathbf{V_1}, \ldots, \mathbf{V_n})$$

a partire dalle proposizione atomiche $\mathbf{V_1}, \ldots \mathbf{V_n}$ associamo una funzione

$$\mathtt{Tab}_{\mathbf{comp}(\mathbf{V_1},\ldots,\mathbf{V_n})} \; : \; \{0,1\}^n \longrightarrow \{0,1\}$$

rappresentata dalla tabella di verità

V_1	V_2	...	V_n	$\mathbf{comp}(\mathbf{V_1}, \ldots, \mathbf{V_n})$
0	1	...	...	$\mathbf{c_1}$
0	0	...	...	$\mathbf{c_2}$
1	1	...	...	$\mathbf{c_3}$
1	0	......	...	...
...	...	...	...	...
...	...	...	...	...

che associa a $\mathbf{comp}(\mathbf{V_1}, \ldots, \mathbf{V_n})$ un valore IN USCITA $\mathbf{c_i}$ che può solo essere **1** (per **vero**) oppure **0** (per **falso**) al variare delle combinazioni di valori **0** e **1** associate alle proposizioni atomiche $\mathbf{V_i}$ per $\mathbf{i = 1, \ldots, n}$.

6.7.1 Come si costruisce la tabella di verità di una proposizione?

La tabella di ogni **proposizione formale pr** si costruisce **componendo** (come funzioni) le tabelle dei connettivi

$$\neg\,,\ \vee\,,\ \&\,,\ \rightarrow$$

che compongono **pr** e che definiamo di seguito.

6.7.2 Tabella di verità di ¬

si ottiene considerando che

$$\neg\mathbf{A} \quad \textbf{è vero} \qquad \text{sse} \qquad \mathbf{A} \quad \textbf{è falso}$$

ed è la funzione unaria

A	$\neg\mathbf{A}$
0	1
1	0

6.7.3 Tabella di verità di &

si ottiene considerando che

A&B **è vero** sse **A** **è vero** **e** **B** **è vero**

ed è la funzione binaria

A	B	$A\&B$
0	1	0
0	0	0
1	1	1
1	0	0

6.7.4 Tabella di verità di $\vee$

si ottiene considerando che

A$\vee$B **è vero** **sse** **A** **è vero** **o** **B** **è vero**
o **sono veri entrambi**

ed è la funzione binaria

A	**B**	**A$\vee$B**
0	1	1
0	0	0
1	1	1
1	0	1

6.7.5 Tabella di verità di $\rightarrow$

si ottiene considerando che

A$\rightarrow$B **è vero** sse **$\neg$A$\vee$B** **è vero**

ed è la funzione binaria

A	**B**	**$\neg$A**	**A$\rightarrow$B**
0	1	1	1
0	0	1	1
1	1	0	1
1	0	0	0

6.7.6 VALIDITÀ CLASSICA di una proposizione

Una proposizione `pr` è **vera classicamente** sse **la tabella di verità di `pr` dà sempre** 1 **in uscita** e sinonimi di "`pr` è **vera classicamente**" sono i seguenti:

- "`pr` è **TAUTOLOGIA classica**"
- "`pr` è **VALIDA classicamente**"
- la scrittura simbolica $\models$ `pr` che si legge "`pr` **vale classicamente**"

6.7.7 Esempio di tabella di verità

La tabella di verità di (**A**→**B**)∨**A** si ottiene costruendo dapprima la tabella di $A \rightarrow B$ e poi combinandola con la disgiunzione con A come segue

A	**B**	**A**→**B**	(**A**→**B**)∨**A**
0	1	1	1
0	0	1	1
1	1	1	1
1	0	0	1

6.8 Proposizioni VALIDE, SODDISFACIBILI e i loro NON

Una proposizione `pr` si dice

VALIDA in logica classica se è una tautologia, ovvero se sono 1 TUTTI i valori in uscita della sua tabella di verità

SODDISFACIBILE se è 1 QUALCHE valore in uscita della sua tabella di verità

NON VALIDA se è 0 QUALCHE valore in uscita della sua tabella di verità

INSODDISFACIBILE o **PARADOSSO** se sono 0 TUTTI i valori in uscita della sua tabella di verità.

esempio di proposizione VALIDA: ¬¬(**P**∨¬**P**)

esempio di proposizione SODDISFACIBILE: ¬**P** per $\mathbf{P} = \mathbf{0}$

esempio di proposizione NON VALIDA: **P**→¬**P** per $\mathbf{P} = \mathbf{1}$

esempio di proposizione INSODDISFACIBILE: **P**&¬**P**

6.8.1 Esempi di analisi validità proposizioni

1. (**A**→**B**)&**A** è una tautologia? Ovvero vale $\models$ (**A**→**B**)&**A** ?

   Se facciamo la tabella di verità per (**A**→**B**)&**A** otteniamo

A	**B**	**A**→**B**	(**A**→**B**)&**A**
0	1	1	0
0	0	1	0
1	1	1	1
1	0	0	0

   e concludiamo che è NON VALIDA (per esempio per $A = B = 0$) $\Rightarrow$ NON è tautologia $\Rightarrow$ NON vale $\models$ (**A**→**B**)&**A**
   Concludiamo però che è soddisfacibile per $A = B = 1$.

2. Guardando la tabella di (**A**→**B**)&**A** concludete che $\models$ ¬((**A**→**B**)&**A**) vale ???

   Chiaramente ¬((**A**→**B**)&**A**) è NON VALIDA (per i valori $A = B = 1$ ad esempio) e SODDISFACIBILE (per i valori $A = B = 0$).

3. vale

$$\models (\mathbf{A}\rightarrow\mathbf{B})\vee\mathbf{A} \quad ?$$

Se facciamo la tabella di verità

A	**B**	**A→B**	**(A→B)∨A**
0	1	1	1
0	0	1	1
1	1	1	1
1	0	0	1

otteniamo che la formula $(\mathbf{A}\rightarrow\mathbf{B})\vee\mathbf{A}$ è **VERA classicamente**, $\Rightarrow$ è una **tautologia classica** $\Rightarrow$ vale $\models (\mathbf{A}\rightarrow\mathbf{B})\vee\mathbf{A}$. Chiaramente la sua negazione $\neg(\,(\mathbf{A}\rightarrow\mathbf{B})\vee\mathbf{A}\,)$ è INSODDISFACIBILE.

6.9 UGUAGLIANZA tra proposizioni

L'identità sintattica NON è il concetto più rilevante di uguaglianza tra proposizioni. L'uguaglianza tra proposizioni che ci interessa è quella che *identifica due proposizioni come* **uguali** *se hanno la stessa tabella di verità*. E quindi ci chiediamo:

quando due proposizioni formali $\mathbf{pr_1}$ *e* $\mathbf{pr_2}$ *hanno la STESSA tabella di verità?*

Innanzitutto notiamo che proposizioni sintatticamente diverse possono avere la stessa tabella di verità: per esempio

A	$\neg\mathbf{A}$
0	1
1	0

è anche la tabella di verità per $\neg\mathbf{A}\&\neg\mathbf{A}$

A	$\neg A\&\neg A$
0	1
1	0

e anche per $(\,\neg\mathbf{A}\&\neg\mathbf{A}\,)\&\neg\mathbf{A}$ e per $(\,(\,\neg\mathbf{A}\&\neg\mathbf{A}\,)\&\neg\mathbf{A}\,)\&\neg\mathbf{A}$.

Per capire come sono relazionate tale proposizioni introduciamo il connettivo di equivalenza (o equiprovabilità).

6.9.1 Connettivo equivalenza

Indichiamo con il segno

$$\leftrightarrow$$

il connettivo **equivalenza** che è definito in tal modo: date due proposizioni formali $\mathtt{pr}_1$ e $\mathtt{pr}_2$

$$\mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2 \;\equiv\; (\,\mathtt{pr}_1\rightarrow\mathtt{pr}_2\,)\ \&\ (\,\mathtt{pr}_2\rightarrow\mathtt{pr}_1\,)$$

che si legge "$\mathtt{pr}_1$ è **equivalente** a "$\mathtt{pr}_2$".

Il connettivo "equivalenza" ha quindi la seguente tabella di verità

A	**B**	**A$\leftrightarrow$B**
0	1	0
0	0	1
1	1	1
1	0	0

Questo connettivo è importante perchè cattura esattamente l'uguaglianza semantica delle tabelle di verità:

Theorem 6.1 *Date proposizioni* $\mathtt{pr}_1$ *e* $\mathtt{pr}_2$*, allora*

$\mathtt{pr}_1$ *e* $\mathtt{pr}_2$ *hanno la* **stessa tabella di verità** *(contenente tutte le variabili che compaiono in entrambe)*

sse vale

$\models \mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2$

e in tal caso si dice che la proposizione $\mathtt{pr}_1$ è *uguale semanticamente* a $\mathtt{pr}_2$, ovvero l'**uguaglianza semantica** di proposizioni è l' **equivalenza** di proposizioni.

Chiaramente se due proposizioni sono equivalenti, allora una è una tautologia sse lo è anche l'altra visto che hanno la stessa tabella di verità:

Corollary 6.2 *Date proposizioni* $\mathtt{pr}_1$ *e* $\mathtt{pr}_2$*, allora*

se $\models \mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2$
allora
$\models \mathtt{pr}_1$ *sse* $\models \mathtt{pr}_2$

6.9.2 Precisazione sull'identità sintattica

Precisiamo che consideriamo il connettivo dell'equivalenza $\mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2$ come **ABBREVIAZIONE** di

$$(\, \mathtt{pr}_1 \rightarrow \mathtt{pr}_2 \,)\ \&\ (\, \mathtt{pr}_2 \rightarrow \mathtt{pr}_1 \,)$$

Quindi diciamo che la proposizione $\mathbf{pr_1}$ è **uguale sintatticamente** a $\mathbf{pr_2}$ se le due proposizioni soddisfano una delle seguenti condizioni:

- sono proprio scritte nello stesso modo;
- $\mathbf{pr_1}$ è ottenuta da $\mathbf{pr_2}$ sostituendo i nomi abbreviati con il loro significato.

Per esempio

$$A\&(B \leftrightarrow C) \text{ è uguale sintatticamente a } A\&(\ (B \rightarrow C)\ \&\ (C \rightarrow B)\ )$$

Ovviamente proposizioni *sintatticamente uguali* sono anche *semanticamente uguali*, ovvero *hanno la stessa tabella di verità*!!

6.10 Tautologie classiche

Di seguito diamo una lista di proposizioni valide classicamente e lasciamo al lettore verificare che la loro tabella di verità ha TUTTI 1 in uscita:

essenza implicazione	$\models(\ \mathbf{A}\rightarrow\mathbf{B}\ )\leftrightarrow\neg\mathbf{A}\vee\mathbf{B}$
associatività $\vee$	$\models(\ \mathbf{A}\vee\mathbf{B}\ )\vee\mathbf{C}\leftrightarrow\mathbf{A}\vee(\ \mathbf{B}\vee\mathbf{C}\ )$
associatività &	$\models(\ \mathbf{A}\&\mathbf{B}\ )\&\mathbf{C}\leftrightarrow\mathbf{A}\&(\ \mathbf{B}\&\mathbf{C}\ )$
commutatività $\vee$	$\models\mathbf{A}\vee\mathbf{B}\leftrightarrow\mathbf{B}\vee\mathbf{A}$
commutatività &	$\models\mathbf{A}\&\mathbf{B}\leftrightarrow\mathbf{B}\&\mathbf{A}$
distributività $\vee$ su &	$\models\mathbf{A}\vee(\ \mathbf{B}\&\mathbf{C}\ )\leftrightarrow(\ \mathbf{A}\vee\mathbf{B}\ )\&(\ \mathbf{A}\vee\mathbf{C}\ )$
distributività & su $\vee$	$\models\mathbf{A}\&(\ \mathbf{B}\vee\mathbf{C}\ )\leftrightarrow(\ \mathbf{A}\&\mathbf{B}\ )\vee(\ \mathbf{A}\&\mathbf{C}\ )$
idempotenza $\vee$	$\models\mathbf{A}\vee\mathbf{A}\leftrightarrow\mathbf{A}$
idempotenza &	$\models\mathbf{A}\&\mathbf{A}\leftrightarrow\mathbf{A}$
leggi di De Morgan	$\models\neg(\ \mathbf{B}\vee\mathbf{C}\ )\leftrightarrow\neg\mathbf{B}\&\neg\mathbf{C}$
	$\models\neg(\ \mathbf{B}\&\mathbf{C}\ )\leftrightarrow\neg\mathbf{B}\vee\neg\mathbf{C}$
legge della doppia negazione	$\models\neg\neg\mathbf{A}\leftrightarrow\mathbf{A}$
legge della NON contraddizione	$\models\neg(\ \mathbf{A}\&\neg\mathbf{A}\ )$
legge del terzo escluso	$\models\mathbf{A}\vee\neg\mathbf{A}$

In ogni tautologia descritta sopra

$$\mathtt{pr}_1(A,B,C)$$

con variabili $A,B,C...$ si ottiene una nuova tautologia

$$\mathtt{pr}_1(\mathtt{pr}_2,\mathtt{pr}_3,\mathtt{pr}_4)$$

se si sostituiscono A,B,C in $\mathtt{pr}_1(A,B,C)$ con proposizioni arbibrarie rispettivamente $\mathtt{pr}_2,\mathtt{pr}_3,\mathtt{pr}_4$.

Una prova di ciò è costituita dal fatto che la tabella finale della proposizione $\mathtt{pr}_1(\mathtt{pr}_2,\mathtt{pr}_3,\mathtt{pr}_4)$ è composizione di una tabella di una tautologia, che è funzione costante 1, con le tabelle delle proposizioni sostituite, ma siccome comporre con una funzione costante 1 dà luogo ancora ad una funzione costante 1, ne segue che la proposizione $\mathtt{pr}_1(\mathtt{pr}_2,\mathtt{pr}_3,\mathtt{pr}_4)$ è pure tautologia.

Ad esempio è pure tautologia classica

$$\models\mathtt{pr}\vee\neg\mathtt{pr}$$

ove **pr** è una qualsiasi altra proposizione.

La chiusura per sostituzione delle tautologie si può esprimere in tal modo:

Teorema di sostituzione semplice
Date le proposizioni $\mathtt{pr}_1(\mathtt{A})$ e $\mathtt{pr}_2$

$$\boxed{\begin{array}{c}\text{Se} \models \mathtt{pr}_1(\mathtt{A})\\ \text{allora}\\ \models\mathtt{pr}_1(\mathtt{pr}_2)\end{array}}$$

6.11 Proprietà utili sull'equivalenza

Ricordando che **proposizioni equivalenti** hanno **stessa tabella di verità** si ottiene che se è una vera pure l'altra lo è:

Proposition 6.3 (equivalenti/verità) *Date* $\mathtt{pr}_1$ *e* $\mathtt{pr}_2$ *proposizioni*

se $\models \mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2$
allora $\models \mathtt{pr}_1$ *sse* $\models \mathtt{pr}_2$

Ora mostriamo che l'essere equivalenti è una relazione simmetrica e transitiva:

Lemma 6.4 (simmetria equivalenti) *Date* $\mathtt{pr}_1$ *e* $\mathtt{pr}_2$ *proposizioni*

se $\models \mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2$ *allora* $\models \mathtt{pr}_2 \leftrightarrow \mathtt{pr}_1$

Esempio: per dedurre che $\models A \leftrightarrow A\&A$ vale basta usare la simmetria dell'equivalenza sopra a partire dall'idempotenza della & in sezione 6.10.

Lemma 6.5 (transitività equivalenti) *Date* $\mathtt{pr}_1$, $\mathtt{pr}_2$ *e* $\mathtt{pr}_3$ *proposizioni*

se $\models \mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2$ *e* $\models \mathtt{pr}_2 \leftrightarrow \mathtt{pr}_3$
allora $\models \mathtt{pr}_1 \leftrightarrow \mathtt{pr}_3$

Esempio: per dedurre che $\models A \vee A \leftrightarrow A\&A$ vale basta usare la transitività dell'equivalenza a partire dalla simmetrica dell'idempotenza della & in sezione 6.10 ovvero $A \leftrightarrow A\&A$ e dall'idempotenza della $\vee$.

Ora si noti che se due proposizioni sono uguali a meno di un loro pezzo e i due pezzi diversi sono equivalenti, allora le due proposizioni iniziali sono equivalenti. Per esempio le proposizioni

$$(C \to D)\&B \qquad\qquad (\neg C \vee D)\&B$$

hanno i pezzi $C \to D$ e $\neg C \vee D$ che sono equivalenti per l'essenza della implicazione in sezione 6.10 e quindi loro sono equivalenti. Questa è un'istanza del seguente teorema:

Theorem 6.6 (equivalenza per pezzi) *Se vale* $\models \mathtt{pr}_1 \leftrightarrow \mathtt{pr}_2$*, ovvero* $\mathtt{pr}_1$ *è equivalente a* $\mathtt{pr}_2$

presa un'altra proposizione $\mathbf{pr_3(A)}$ *che è una scrittura per indicare che nella proposizione* $\mathbf{pr_3(A)}$ *compare la variabile* $\mathbf{A}$ *allora vale*

$$\models \mathbf{pr_3(pr_1)} \leftrightarrow \mathbf{pr_3(pr_2)}$$

Per applicare questo teorema al fine di dedurre che $(C \to D)\&B$ è equivalente a $(\neg C \vee D)\&B$ basta considerare la proposizione

$$A\&B$$

e sostituire A una volta con $C \to D$ e si ottiene (dopo aver messo le parentesi) $(C \to D)\&B$ e un'altra volta sostituire A con $\neg C \vee D$ e si ottiene $(\neg C \vee D)\&B$ che è equivalente a $(C \to D)\&B$ per il teorema enunciato.

Dal teorema 6.6 e teorema 6.1 segue come corollario che se in una proposizione $\mathtt{pr_3}$ (A) si sostituisce A una volta con $\mathtt{pr_1}$ e un'altra volta con un suo equivalente $\mathtt{pr_2}$ si ottengono due proposizioni che hanno la proprietà che una è una tautologia sse lo è anche l'altra:

Corollary 6.7 (verità su equivalenza per pezzi) *Se vale* $\models \mathtt{pr_1} \leftrightarrow \mathtt{pr_2}$*, ovvero* $\mathtt{pr_1}$ *è equivalente a* $\mathtt{pr_2}$

presa un'altra proposizione $\mathbf{pr_3(A)}$ *che è una scrittura per indicare che nella proposizione* $\mathbf{pr_3(A)}$ *compare la variabile* $\mathbf{A}$ *allora vale*

$$\models \mathbf{pr_3(pr_1)} \qquad \text{sse } \models \mathbf{pr_3(pr_2)}$$

7 Approfondimento sulle tabelle di verità

Di seguito riportiamo alcuni fatti interessanti relativi alle tabelle di verità.

7.1 Ogni tabella a valori in 0 e 1 è tabella di una proposizione?

Abbiamo visto come ogni proposizione formale possegga una tabella di verità. Ora ci chiediamo:

è anche vero che `ogni` **funzione a valori in** $\{0, 1\}$ *rappresentata da una tabella ad* n *entrate* (con un numero di righe pari alle possibili combinazioni n-arie di 0 e 1) del tipo

V_1	V_2	...	V_n	???
0	1	...	...	$\mathbf{c_1}$
0	0	...	...	$\mathbf{c_2}$
1	1	...	...	$\mathbf{c_3}$
1	0	......	...	...
...	...	...	...	...
...	...	...	...	...

(ove $\mathbf{c_i}$ può essere solo 0 o 1)

corrisponde ad una **proposizione formale** *con (al più) n variabili proposizionali? Se sì dobbiamo forse aggiungere qualche connettivo a quelli già definiti per rappresentarla?*

La risposta è che OGNI TABELLA a n entrate CORRISPONDE alla TABELLA di VERITÀ di una PROPOSIZIONE formale con al più n variabili proposizionali e che NON abbiamo bisogno di aggiungere nuove proposizioni per rappresentare tutte le tabelle di verità.

Il motivo è che vale il seguente teorema:

Theorem 7.1 (Completezza delle tabelle rispetto a $\neg, \vee, \&$**)** *Ogni tabella con* $\mathbf{n}$*-entrate denota un connettivo* $\mathbf{n}$*-ario che si può scrivere con solo* $\vee, \&$ *ed* $\neg$.

Questo teorema è in verità il corollario di altri due teoremi:

Theorem 7.2 (forma normale disgiuntiva) *Ogni tabella con* $\mathbf{n}$*-entrate, (*$\mathbf{1 \leq n}$*) denota un connettivo* $\mathbf{n}$*-ario che si può scrivere in forma normale* **disgiuntiva**

$$\bigvee_{\mathbf{i}\ \textit{indice riga con risultato 1}} \mathbf{riga_i}$$

ove

$$\mathbf{riga_i} \equiv (((\mathbf{C_{i,1}} \& \mathbf{C_{i,2}}) \ldots \& \mathbf{C_{i,n}}$$

è congiunzione di variabili o loro negazioni e

$$\bigvee_{\mathbf{i} \textit{ indice riga con risultato 1}} \mathbf{riga_i} \equiv ((\mathbf{riga_{i_1}} \vee \mathbf{riga_{i_2}}) \vee \mathbf{riga_{i_3}} \ldots) \vee \mathbf{riga_{i_n}}$$

La procedura per scrivere la forma normale disgiuntiva di una tabella di verità a n entrate è la seguente:

- considero la tabella di verità di $\mathbf{conn(V_1, \ldots, V_n)}$
- se **NON ESISTE almeno una riga con risultato 1** poni

$$\mathbf{V_1} \& \neg \mathbf{V_1}$$

- se **ESISTE almeno una riga con risultato 1** faccio la disgiunzione

$$\bigvee_{\mathbf{i} \text{ indice riga con risultato 1}} \mathbf{riga_i}$$

  ove

$$\mathbf{riga_i} \equiv (((\mathbf{C_{i,1}} \& \mathbf{C_{i,2}}) \ldots \& \mathbf{C_{i,n}}$$

  è multipla congiunzione di $\mathbf{C_{i,k}}$ definiti come segue

$$\mathbf{C_{i,k}} \equiv \begin{cases} \mathbf{V_k} & \text{se } \mathbf{1} \text{ è il valore di } \mathbf{V_k} \text{ nella riga } \mathbf{i}\text{-esima} \\ \neg \mathbf{V_k} & \text{se } \mathbf{0} \text{ è il valore di } \mathbf{V_k} \text{ nella riga } \mathbf{i}\text{-esima} \end{cases}$$

- si dimostra che

$$\models \mathbf{conn(V_1, \ldots, V_n)} \quad \leftrightarrow \quad \bigvee_{\mathbf{i} \text{ indice riga con risultato 1}} \mathbf{riga_i}$$

7.1.1 Esempio di uso di forma normale disgiuntiva

Data la tabella di verità

A	B	$\mathbf{conn(A, B)}$
0	1	0
0	0	1
1	1	0
1	0	0

per scoprire che proposizione è $\mathbf{conn(A, B)}$ usiamo il teorema **forma normale disgiuntiva** e scriviamo dunque le righe uscenti con 1

$$\neg \mathbf{A} \& \neg \mathbf{B}$$

e dal teorema deduciamo che possiamo definire

$$\mathbf{conn(A, B)} \equiv \neg \mathbf{A} \& \neg \mathbf{B}$$

perchè connettivi equivalenti hanno la stessa tabella di verità.

Se prendiamo invece questa tabella di verità

A	B	$\mathbf{conn(A,B)}$
0	1	1
0	0	1
1	1	0
1	0	1

chi è $\mathbf{conn(A,B)}$? Per stabilirlo di nuovo usiamo il teorema di forma normale disgiuntiva e scriviamo dunque le righe uscenti con 1:

$$(\ (\ \neg\mathbf{A}\&\mathbf{B}\ )\vee(\neg\mathbf{A}\&\neg\mathbf{B})\ )\vee(\ \mathbf{A}\&\neg\mathbf{B}\ )$$

dal teorema sappiamo che possiamo definire

$$\mathbf{conn(A,B)}\ \equiv\ (\ (\ \neg\mathbf{A}\&\mathbf{B}\ )\vee(\neg\mathbf{A}\&\neg\mathbf{B})\ )\vee(\ \mathbf{A}\&\neg\mathbf{B}\ )$$

Però la scrittura del connettivo è molto complessa... Vediamo allora un'altro modo di scrivere il connettivo corrispondente ad una tabella di verità nel caso ci siano pochi 0 in uscita. A tal fine enunciamo il seguente teorema:

Theorem 7.3 (forma normale congiuntiva) *Ogni tabella con* $\mathbf{n}$*-entrate,* $\mathbf{1 \leq n}$*, denota un connettivo* $\mathbf{n}$*-ario che si può scrivere in forma normale* **congiuntiva**

$$\&_{\mathbf{i}\ \textit{riga con risultato 0}}\ \overline{\mathbf{riga_i}}$$

ove

$$\overline{\mathbf{riga_i}} \equiv (((\ \mathbf{C_{i,1}}\vee\mathbf{C_{i,2}}\ )\ldots\vee\mathbf{C_{i,n}}$$

è disgiunzione di variabili o loro negazioni e

$$\&_{\mathbf{i}\ \textit{riga con risultato 0}}\ \overline{\mathbf{riga_i}}\ \equiv\ (\ (\ \overline{\mathbf{riga_{i_1}}}\&\overline{\mathbf{riga_{i_2}}}\ )\&\overline{\mathbf{riga_{i_3}}}\ldots)\&\overline{\mathbf{riga_{i_n}}}$$

La procedura per scrivere la forma normale congiuntiva di una tabella di verità ad n entrate è la seguente:

- considero la tabella di verità del connettivo **n-ario** $\mathbf{conn(V_1,\ldots,V_n)}$
- se **NON ESISTE almeno una riga con risultato 0** poni

$$\mathbf{V_1}\vee\neg\mathbf{V_1}$$

- se **ESISTE almeno una riga con risultato 0** faccio la la congiunzione

$$\&_{\mathbf{i}\ \text{riga con risultato 0}}\ \overline{\mathbf{riga_i}}$$

  ove

$$\overline{\mathbf{riga_i}}\ \equiv\ (((\ \mathbf{C_{i,1}}\vee\mathbf{C_{i,2}}\ )\ldots\vee\mathbf{C_{i,n}}$$

  è multipla disgiunzione di $\mathbf{C_i}$ definiti come segue

$$\mathbf{C_{i,k}} \equiv \begin{cases} \mathbf{V_k} & \text{se } \mathbf{0} \text{ è il valore di } \mathbf{V_k} \text{ nella riga } \mathbf{i}\text{-esima} \\ \neg\mathbf{V_k} & \text{se } \mathbf{1} \text{ è il valore di } \mathbf{V_k} \text{ nella riga } \mathbf{i}\text{-esima} \end{cases}$$

- si dimostra che

$$\models\mathbf{conn(V_1,\ldots,V_n)}\ \leftrightarrow\ \&_{\mathbf{i}\ \text{riga con risultato 0}}\ \overline{\mathbf{riga_i}}$$

7.1.2 Esempio di uso di forma normale congiuntiva: il connettivo NAND

Quindi ora data la tabella

A	B	$\mathbf{conn(A,B)}$
0	1	1
0	0	1
1	1	0
1	0	1

usiamo il teorema di forma normale congiuntiva e scriviamo le righe uscenti con 0: $\neg\mathbf{A}\vee\neg\mathbf{B}$ e deduciamo dal teorema che possiamo definire

$$\mathbf{conn(A,B)} \equiv \neg\mathbf{A}\vee\neg\mathbf{B}$$

perchè connettivi equivalenti hanno la stessa tabella di verità.

Inoltre per simmetria dell'equivalenza e per la legge di de Morgan in sezione 6.10 otteniamo che

$$\models \mathbf{conn(A,B)} \leftrightarrow \neg(\ \mathbf{A}\&\mathbf{B}\ )$$

e quindi la tabella di verità rappresenta il connettivo NAND.

7.1.3 Raffinamento del teorema di completezza delle tabelle di verità

Theorem 7.4 **(& + ¬)** *Ogni tabella con* **n**-*entrate denota un connettivo* **n**-*ario che si può scrivere con solo & e ¬.*

Dim. Segue per il teorema di forma normale congiuntiva, dopo aver notato che la disgiunzione tramite la legge di De Morgan e quella della doppia negazione in sezione 6.10 si può definire come segue

$$\mathbf{A} \vee \mathbf{B} \ \equiv\ \neg(\neg\mathbf{A}\&\neg\mathbf{B})$$

In particolare, si noti che l'implicazione tramite la sua essenza in sezione 6.10 si può definire in tal modo

$$\mathbf{A} \rightarrow \mathbf{B} \ \equiv\ \neg\mathbf{A} \vee \mathbf{B}$$

Theorem 7.5 **(∨ + ¬)** *Ogni tabella con* **n**-*entrate denota un connettivo* **n**-*ario che si può scrivere con solo ∨ e ¬.*

Dim: Segue per il teorema di forma normale disgiuntiva dopo aver notato che la congiunzione tramite le leggi di De Morgan e quella della doppia negazione si può definire come segue

$$\mathbf{A}\&\mathbf{B} \ \equiv\ \neg(\ \neg\mathbf{A}\vee\neg\mathbf{B}\ )$$

Theorem 7.6 **(→ +¬)** *Ogni tabella con* **n**-*entrate denota un connettivo* **n**-*ario che si può scrivere con solo → e ¬.*

Dim: Basta notare che si può definire la disgiunzione in tal modo

$$\mathbf{A} \vee \mathbf{B} \ \equiv\ \neg\mathbf{A} \rightarrow \mathbf{B}$$

grazie alla legge della doppia negazione in sezione 6.10 e poi si applica il teorema 7.5.

Theorem 7.7 **(solo NAND)** *Ogni connettivo* **n-ario** *si può scrivere con solo NAND.*

Dim: Basta notare che tramite **NAND** si può definire sia la negazione che la disgiunzione come segue

$$\neg\mathbf{A} \ \equiv\ \mathbf{NAND(A,A)} \qquad\qquad \mathbf{A} \vee \mathbf{B} \ \equiv\ \mathbf{NAND}(\neg A, \neg B)$$

ove nel secondo si usa ovviamente la definizione di negazione data nella definizione di sinistra. Poi si conclude per il teorema 7.5.

7.1.4 Quante sono le tabelle di verità ad n entrate?

Le possibili tabelle di verità con **n** entrate sono

$$\mathbf{2^{2^n}}$$

ovvero tante quante le funzioni da $\{0,1\}^{\mathbf{n}} \rightarrow \{0,1\}$ e quindi i connettivi **n**-ari **a meno di equivalenza proposizionale** sono $\mathbf{2^{2^n}}$

Per esempio le tabelle di verità **unarie** sono $\mathbf{4} = \mathbf{2^{2^1}}$ e sono:

identità

A	A
0	0
1	1

negazione

A	$\neg A$
0	1
1	0

costante falso

A	$\perp$
0	0
1	0

costante vero

A	$\top$
0	1
1	1

ove $\perp$ è il nome alla costante "falso" che è da aggiungere alle proposizioni, $\top$ è il nome alla costante "vero" che è da aggiungere alle proposizioni.

Si noti che per i teoremi di completezza delle tabelle con i vari linguaggi si deduce che:

$$\models \perp \ \leftrightarrow \ \mathbf{A} \,\&\, \neg\mathbf{A} \qquad\qquad \models \top \ \leftrightarrow \ \mathbf{A} \ \vee \ \neg\mathbf{A}$$

8 Due strategie per verificare una tautologia

Per quanto spiegato finora per vedere se vale

$$\models \mathtt{pr}$$

abbiamo almeno due possibilità:

1. **strategia tabella**: *fai la tabella di verità* di `pr`
   vantaggio: strategia sicura e automatica
   svantaggio: la tabella può essere molto complessa

2. **strategia riduzione**: *riduci* `pr` *tramite equivalenze note ad una tautologia nota*
   vantaggio: strategia veloce, se termina
   svantaggio: strategia non automatica e non sempre terminante in una proposizione nota

Suggerimento: combinate le due strategie sopra!!

Per esempio per vedere se vale

$$\models (\ \mathbf{A} \rightarrow \mathbf{B}\ ) \vee (\mathbf{B} \rightarrow \mathbf{A})$$

si usa due volte

$$\text{essenza} \rightarrow \qquad\qquad \models (\ \mathbf{A} \rightarrow \mathbf{B}\ ) \leftrightarrow (\ \neg\mathbf{A} \vee \mathbf{B}\ )$$

e si ottiene

$$\models (\ \neg\mathbf{A} \vee \mathbf{B}\ ) \vee (\ \neg\mathbf{B} \vee \mathbf{A}\ )$$

Infine per associatività e commutatività di $\vee$ si ottiene

$$\models(\neg\mathbf{A} \vee \mathbf{A}) \vee (\neg\mathbf{B} \vee \mathbf{B})$$

e ora la tabella di verità risulta facile da costruire e dà sempre valore 1 perchè i disgiunti sono entrambe tautologie (la prima compare in sezione 6.10 e la seconda segue per commutatività di $\vee$ dalla legge del terzo escluso.

8.0.5 In logica classica non c'è implicazione causale

La tautologia

$$\models(\mathbf{A}\rightarrow\mathbf{B}) \vee (\mathbf{B}\rightarrow\mathbf{A})$$

mostra con il seguente esempio che l'implicazione della logica classica NON è causale in quanto si trovano delle verità controintuitive riguardanti le implicazioni. Infatti ponendo
A="Voi passerete l'esame di logica"
B="Avete una zia con i calli"
si ottiene che

"**Se voi passerete l'esame di logica allora avete una zia con i calli, oppure se avete una zia con i calli allora passerete l'esame di logica**"

è vera logicamente secondo la logica classica.

8.0.6 Esempio su validità e soddisfacibilità e i loro NON

Esempio: formalizzare in un unica proposizione l'asserzione

"È vero che **se i viaggiatori non sono contenti allora il treno è in ritardo** se si assume che **se i viaggiatori son contenti allora il treno non è in ritardo**."

usando V="**i viaggiatori sono contenti**"
R="**il treno è in ritardo**"
e mostrare se la proposizione ottenuta è tautologia classica e in caso contrario dire per quali valori delle variabili non è valida e se è soddisfacibile (e per quali valori delle variabili lo è) o insoddisfacibile.

La sua formalizzazione come UNICO enunciato è

$$(V \rightarrow \neg R) \rightarrow (\neg V \rightarrow R)$$

Usando il teorema 6.6 sull'essenza dell'implicazione due volte otteniamo che

$$\models (\ (V \rightarrow \neg R) \rightarrow (\neg V \rightarrow R) \ ) \ \leftrightarrow \ (\ \neg V \vee \neg R \rightarrow \neg\neg V \vee R \ )$$

Di nuovo usando il teorema 6.6 sulla legge della doppia negazione otteniamo che

$$\models (\ \neg V \vee \neg R \rightarrow \neg\neg V \vee R \ ) \ \leftrightarrow \ (\ \neg V \vee \neg R \rightarrow V \vee R \ )$$

e per transitività si deduce

$$\models (\ (V \rightarrow \neg R) \rightarrow (\neg V \rightarrow R) \ ) \ \leftrightarrow \ (\ \neg V \vee \neg R \rightarrow V \vee R \ )$$

Ora si vede che $\neg V \vee \neg R \rightarrow V \vee R$ NON è VALIDO falsificando la conseguenza $V \vee R$ e quindi ponendo $V = R = 0$, da cui si ricava che $\neg V \vee \neg R$ deve essere 1 nella sua tabella per i valori dati. Perciò l'implicazione è falsa per $V = R = 0$ perchè l'antecedente risulta vero su tali valori mentre il conseguente falso.

Si vede poi che $\neg V \vee \neg R \rightarrow V \vee R$ è SODDISFACIBILE perchè basta mettere a 0 i valori dell'antecedente (ovvero porre $\neg V \vee \neg R = 0$) ponendo $V = R = 1$. Per tali valori l'implicazione risulta vera.

Dal fatto che vale $\models$ ($(V \rightarrow \neg R) \rightarrow (\neg V \rightarrow R)$) $\leftrightarrow$ ($\neg V \vee \neg R \rightarrow V \vee R$) ovvero che $(V \rightarrow \neg R) \rightarrow (\neg V \rightarrow R)$ ha la stessa tabella di verità di $\neg V \vee \neg R \rightarrow V \vee R$, i risultati su NON validità e soddisfacibilità ottenuti per il secondo membro dell'equivalenza sopra valgono pure per il primo membro $(V \rightarrow \neg R) \rightarrow (\neg V \rightarrow R)$.