UNITELMA SAPIENZA

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Metodi matematici per l'Informatica

Modulo 13 – Logica proposizionale

Docente: Pietro Cenciarelli

UNITELMA SAPIENZA
SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Logica proposizionale

proposizioni $A, B, \ldots ::= \overbrace{P \mid Q \mid \ldots}^{\text{simboli proposizionali}} \mid \mathit{falso} \mid A \vee B \mid A \wedge B \mid A \rightarrow B \mid \neg A$

- un simbolo proposizionale (P, Q, …) è una proposizione
- *falso* è una proposizione
- se A e B sono proposizioni, allora $A \vee B$ è una proposizione
- se A e B sono proposizioni, allora $A \wedge B$ è una proposizione
- se A e B sono proposizioni, allora $A \rightarrow B$ è una proposizione
- se A è una proposizione, allora $\neg A$ è una proposizione
- *nient'altro è una proposizione*

UNITELMA SAPIENZA
SAPIENZA UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Logica proposizionale

$$\overbrace{\text{P | Q | …}}^{\text{simboli proposizionali}}$$

proposizioni A, B, … ::= P | Q | … | *falso* | A ∨ B | A ∧ B | A → B | ¬ A

modelli m : *simboli proposizionali* → {T, F}

m si estende alle proposizioni (*interpretazione*) come segue: m(*falso*) = F e, per le proposizioni composte, applicando le *tavole di verità*:

A	B	A ∧ B
T	T	T
T	F	F
F	T	F
F	F	F

A	B	A ∨ B
T	T	T
T	F	T
F	T	T
F	F	F

A	B	A → B
T	T	T
T	F	F
F	T	T
F	F	T

A	¬A
T	F
F	T

Logica proposizionale

proposizioni A, B, … ::= $\overbrace{\text{P | Q | …}}^{\text{simboli proposizionali}}$ | *falso* | A ∨ B | A ∧ B | A → B | ¬ A

modelli m : *simboli proposizionali* → {T, F}

m si estende alle proposizioni (*interpretazione*) come segue: m(*falso*) = F
e, per le proposizioni composte, applicando le *tavole di verità*

Se m(A) = T, si dice che m *soddisfa* A e si scrive $\models_m A$

A si dice *soddisfacibile* se esiste almeno un modello che la soddisfa

A è *conseguenza semantica* di $B_1, B_2, \dots B_n$ se è *soddisfatta in ogni modello che soddisfa* $B_1, B_2, \dots B_n$, e si scrive $B_1, B_2, \dots B_n \models A$

A si dice *valida* (una *tautologia*) se è *soddisfatta in ogni modello*, e si scrive $\models A$

Logica proposizionale

$P \rightarrow Q$ *soddisfacibile*

Q *soddisfacibile*

$P \land \neg P$ *insoddisfacibile*

$(A \land B) \rightarrow (A \lor B)$ *valida*

A	B	$A \land B$	$A \lor B$	$(A \land B) \rightarrow (A \lor B)$
T	T	T	T	T
T	F	F	T	T
F	T	F	T	T
F	F	F	F	T

Logica proposizionale

$P \rightarrow Q$ *soddisfacibile*

Q *soddisfacibile*

$P \wedge \neg P$ *insoddisfacibile*

$(A \wedge B) \rightarrow (A \vee B)$ *valida*

$B \vDash A \rightarrow B$

A	B	A → B
T	$\boxed{T}$	$\boxed{T}$
T	F	F
F	$\boxed{T}$	$\boxed{T}$
F	F	T

Logica proposizionale

$P \rightarrow Q$ *soddisfacibile*

Q *soddisfacibile*

$P \wedge \neg P$ *insoddisfacibile*

$(A \wedge B) \rightarrow (A \vee B)$ *valida*

$B \vDash A \rightarrow B$ *conseguenza semantica*

$A, A \rightarrow B \vDash B$ segue immediatamente dal *teorema di deduzione semantica*

UNITELMA SAPIENZA
SAPIENZA UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Deduzione semantica

Algebre di Boole/Heyting

$A \rightarrow B$ è *il più grande* elemento X tale che

$$X \wedge A \leq B$$

$$C \wedge A \leq B \quad \text{sse} \quad C \leq A \rightarrow B$$

Logica classica/intuizionista

$A \rightarrow B$ è *la più debole* proposizione X tale che

$$X, A \vDash B$$

$$C, A \vDash B \quad \text{sse} \quad C \vDash A \rightarrow B$$

UNITELMA SAPIENZA

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Deduzione semantica

$C_1, C_2, \dots C_n, A \vDash B$ se e solo se $C_1, C_2, \dots C_n \vDash A \rightarrow B$

Si può dimostrare per induzione su n.

UNITELMA SAPIENZA
SAPIENZA Università di Roma
Dipartimento di Informatica

Deduzione semantica

☺

$A \vDash B \quad \Longrightarrow \quad \vDash A \to B$

Si può dimostrare per induzione su n.

Per n = 0 …

Se $A \vDash B$ allora, per ogni m, m(A) = T implica m(B) = T.

Dalla tabella di verità di $\to$ ne consegue che, per ogni m, $m(A \to B) = T$

ovvero $\vDash A \to B$.

Deduzione semantica

☺

$A \vDash B \quad \Longleftarrow \quad \vDash A \rightarrow B$

Si può dimostrare per induzione su n.

Per n = 0 ☺

Se $\vDash A \rightarrow B$ allora, per ogni m, m(A→B) = T .

Dalla tabella di verità di → ne consegue che, per ogni m, se m(A) = T, allora m(B) = T, ovvero $A \vDash B$.

Deduzione semantica

$C_1, C_2, \ldots C_n, A \vDash B$ se e solo se $C_1, C_2, \ldots C_n \vDash A \rightarrow B$

Si può dimostrare per induzione su n.

Per n = 0 ☺. Passo induttivo … *esercizio!*

UNITELMA SAPIENZA
SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Deduzione semantica

conseguenze…

$C_1, C_2, \ldots C_n, A \vDash B$ se e solo se $C_1, C_2, \ldots C_n \vDash A \rightarrow B$

$B \vDash A \rightarrow B$ *conseguenza semantica*

$A, A \rightarrow B \vDash B$ conseguenza immediata del *teorema di deduzione semantica…*

Deduzione semantica

conseguenze…

$C_1, C_2, \ldots C_n, A \vDash B$ se e solo se $C_1, C_2, \ldots C_n \vDash A \rightarrow B$

$A \rightarrow B \vDash A \rightarrow B \quad \Longrightarrow \quad A, A \rightarrow B \vDash B$

$B \vDash A \rightarrow B$

$A, A \rightarrow B \vDash B$ ☺

Deduzione semantica

conseguenze…

$C_1, C_2, \ldots C_n, A \vDash B$ se e solo se $C_1, C_2, \ldots C_n \vDash A \rightarrow B$

$B, A \vDash B \quad \Rightarrow \quad B \vDash A \rightarrow B$

$\Rightarrow \quad \vDash B \rightarrow (A \rightarrow B)$

$B \vDash A \rightarrow B$ ☺

$A, A \rightarrow B \vDash B$ ☺

Deduzione semantica

conseguenze…

$C_1, C_2, \ldots C_n, A \vDash B$ se e solo se $C_1, C_2, \ldots C_n \vDash A \rightarrow B$

David Hilbert
1862 - 1943

$A \rightarrow A$

$B \rightarrow (A \rightarrow B)$

$(A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow B) \rightarrow (A \rightarrow C)$

$(\neg A \rightarrow \neg B) \rightarrow (B \rightarrow A)$

assiomi della logica
proposizionale classica

(coming soon…)

UNITELMA SAPIENZA
SAPIENZA UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Equivalenza semantica

Due proposizioni A e B si dicono *semanticamente equivalenti* quando, per ogni interpretazione m, m(A) = T sse m(B) = T.

$$A \equiv B$$

Nota: $A \equiv B$ *quando* $A \vDash B$ *e* $B \vDash A$

Equivalenza semantica

Algebre di Boole	Logica (classica)
$A \vee (B \vee C) = (A \vee B) \vee C$	$A \vee (B \vee C) \equiv (A \vee B) \vee C$
$A \wedge (B \wedge C) = (A \wedge B) \wedge C$	$A \wedge (B \wedge C) \equiv (A \wedge B) \wedge C$
$A \vee B = B \vee A$	$A \vee B \equiv B \vee A$
$A \wedge B = B \wedge A$	$A \wedge B \equiv B \wedge A$
$A \vee (A \wedge B) = A$	$A \vee (A \wedge B) \equiv A$
$A \wedge (A \vee B) = A$	$A \wedge (A \vee B) \equiv A$
$A \vee A = A$ $\quad$ $A \wedge A = A$	$A \vee A \equiv A$ $\quad$ $A \wedge A \equiv A$
$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$	$A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$
$A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$	$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$
$A \vee \bot = A$ $\quad$ $A \wedge \top = A$	$A \vee \text{falso} \equiv A$ $\quad$ $A \wedge \text{vero} \equiv A$
$A \vee \overline{A} = \top$ $\quad$ $A \wedge \overline{A} = \bot$	$A \vee \neg A \equiv \text{vero}$ $\quad$ $A \wedge \neg A \equiv \text{falso}$

Equivalenza semantica

$\neg\neg A \equiv A$ *doppia negazione*

$\neg(A \vee B) \equiv \neg A \wedge \neg B$

$\neg(A \wedge B) \equiv \neg A \vee \neg B$ *De Morgan*

$m(\neg(A \vee B)) = T$ sse $m(A \vee B) = F$

sse $m(A) = F$ e $m(B) = F$

sse $m(\neg A) = T$ e $m(\neg B) = T$

sse $m(\neg A \wedge \neg B) = T$

Equivalenza semantica

$\neg\neg A \equiv A$ *doppia negazione*

$m(\neg\neg A) = T$ sse $m(\neg A) = F$ sse $m(A) = T$

vale nella logica classica *ma non in quella intuizionistica*

♥□≠ ¬¬♥□

Equivalenza semantica

$\neg\neg A \equiv A$ *doppia negazione*

$A \vDash \neg\neg A$ e $\neg\neg A \vDash A$

☹

vale nella logica classica *ma non in quella intuizionistica*

$\top$
|
♥
|
$\bot$

$\neg A \stackrel{\text{def}}{=} A \rightarrow \bot$

$\neg\top = \bot$

$\neg♥ = \bot$

$\neg\bot = \top$

$\neg\neg♥ = \top \not< ♥$

$\neg\neg A \nvDash A$

Equivalenza semantica

$\neg\neg A \equiv A$ *doppia negazione*

$A \models \neg\neg A$ e $\neg\neg A \models A$

☺ ☹

vale nella logica classica *ma non in quella intuizionistica*

$$\neg A \stackrel{\text{def}}{=} A \to \bot$$

$A \models \neg\neg A$

$A \models (A \to \bot) \to \bot$ sse $A, (A \to \bot) \models \bot$

Equivalenza semantica

$\neg\neg A \equiv A$ *doppia negazione*

$A \models \neg\neg A$ e $\neg\neg A \models A$

☺ ☹

vale nella logica classica *ma non in quella intuizionistica*

dove $\neg\neg A$ consegue da A, *ma non viceversa*!

Equivalenza semantica

$$A \wedge B \equiv \neg\neg(A \wedge B) \equiv \neg(\neg A \vee \neg B)$$

$\neg\neg A \equiv A$ *doppia negazione*

$\neg(A \vee B) \equiv \neg A \wedge \neg B$

$\neg(A \wedge B) \equiv \neg A \vee \neg B$

De Morgan

Completezza funzionale

$A \land B \equiv \neg(\neg A \lor \neg B)$

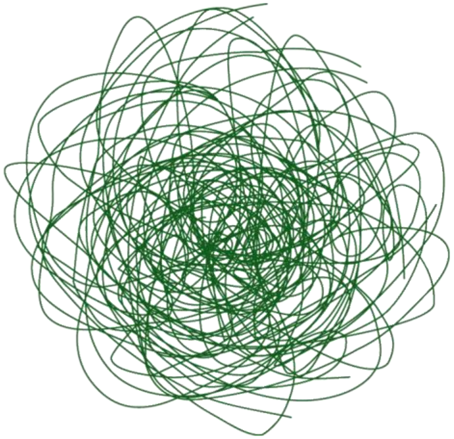
$A \lor B \equiv \neg(\neg A \land \neg B)$

$A \lor B \equiv \neg A \to B$

$A \to B \equiv \neg A \lor B$

$\neg A \equiv A \to \mathit{falso}$

$\mathit{falso} \equiv A \land \neg A$



bastano $\to$ e *falso*

oppure $\lor$ e $\neg$

oppure …

Completezza funzionale

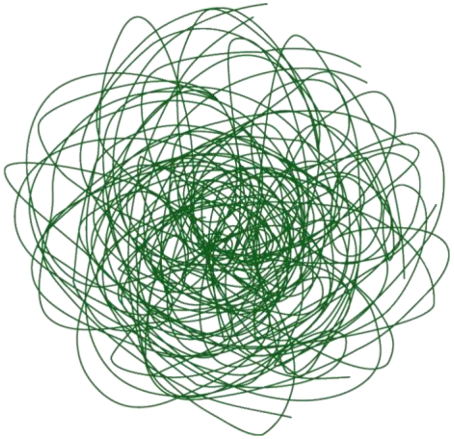
Esistono *altri operatori* logici *non* esprimibili in termini di quelli dati?

$$A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A)$$

$$A \dot{\vee} B \equiv (A \wedge \neg B) \vee (B \wedge \neg A)$$

A	B	$A \dot{\vee} B$ (*XOR*)
T	T	F
T	F	T
F	T	T
F	F	F

verificare!



bastano $\rightarrow$ e *falso*

oppure $\vee$ e $\neg$

oppure …

Completezza funzionale

Esistono *altri operatori* logici *non* esprimibili in termini di quelli dati?

$\circledast$ (A, B, C)

$A \dot{\vee} B \equiv (A \wedge \neg B) \vee (B \wedge \neg A)$

A	B	$A \dot{\vee} B$
T	T	F
T	F	T
F	T	T
F	F	F

A	B	C	$\circledast$ (A, B, C)
T	T	T	T
T	T	F	F
T	F	T	T
T	F	F	F
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

Completezza funzionale

Esistono *altri operatori* logici *non* esprimibili in termini di quelli dati?

$$\circledast(A, B, C) \equiv (A \wedge B \wedge C) \vee (A \wedge \neg B \wedge C) \vee (\neg A \wedge \neg B \wedge C)$$

verificare!

A	B	C	⊛ (A, B, C)
T	T	T	T
T	T	F	F
T	F	T	T
T	F	F	F
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

UNITELMA SAPIENZA
SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Completezza funzionale

Esistono *altri operatori* logici *non* esprimibili in termini di quelli dati?

No!