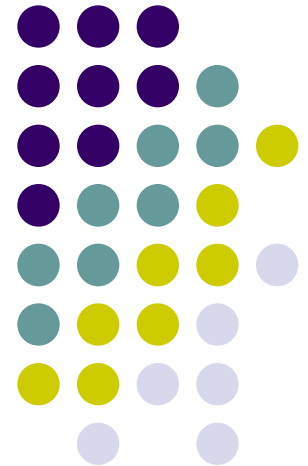
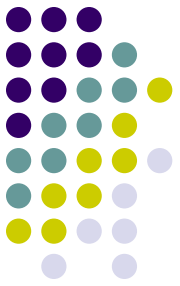


Fondamenti di Informatica

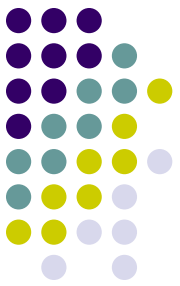
Calcolo Proposizionale



LOGICA

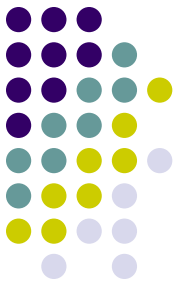


- La logica è la disciplina che studia le regole fondamentali del ragionamento
- **Logica deduttiva**: date le ipotesi (regole generali o fatti inerenti il problema da studiare) si derivano le conclusioni (esprese anch'esse come regole o fatti)
- **Logica induttiva**: a partire da dati derivanti dall'osservazione deriva regole generali



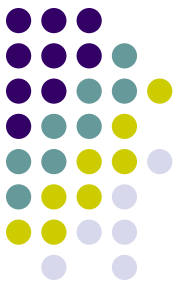
Logica induttiva

- Ragionamento tipico della fisica sperimentale
- Es: derivazione della legge di gravità dall'osservazione sperimentale di fatti empirici (mela di newton)
- Dal “particolare” all’”universale”
- Difficilmente automatizzabile!



Logica deduttiva

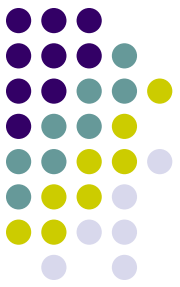
- Ragionamento tipico della matematica
- Es: dimostrazione di teoremi
- Dall' "universale" al "particolare"
- Facilmente **formalizzabile** ed **automatizzabile**



Calcolo proposizionale

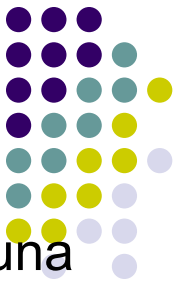
- **Esistono diverse formalizzazioni possibili della logica deduttiva**
- Una di esse è il calcolo proposizionale (o logica enunciativa)
- **Esempio:**
 - Oggi è mercoledì 5 ottobre
 - Proposizione (enunciato) **A: oggi è mercoledì**
 - Proposizione (enunciato) **B: oggi è il 5 ottobre**

Principi base del calcolo proposizionale



- Una proposizione (o condizione, o predicato) può assumere valore logico VERO oppure FALSO
- Una proposizione può essere atomica o composta
- La composizione di proposizioni si effettua mediante operatori detti CONNETTIVI LOGICI
- Il confronto tra entità può essere effettuato mediante OPERATORI DI RELAZIONE
- Il “ragionamento” cioè le argomentazioni mediante le quali dalle proposizioni iniziali si giunge all’enunciato finale sono espresse mediante regole di inferenza

Operatori di relazione



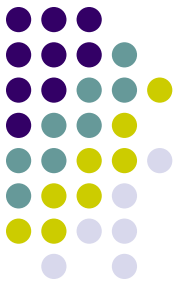
- Gli **operatori di relazione (o di confronto)** permettono di esprimere una proposizione atomica (condizione) confrontando due operandi.
- I più noti sono quelli che confrontano **insiemi numerici**:
 - uguale (simbolo '=')
 - diverso (simbolo '≠')
 - maggiore (simbolo '>')
 - minore (simbolo '<')
 - maggiore o uguale (simbolo '≥')
 - minore o uguale (simbolo '≤')
- Il **risultato di un confronto** assume valore **Vero** o **Falso** (**ovvero, un valore logico**) pertanto lo possiamo comporre in predicati:
 - $5 < 3$? Falso
 - $-1 > -5$? Vero
- Si noti che non tutti gli operatori di relazione possono essere applicati a valori di qualunque natura

Connettivi logici



- I connettivi o operatori traducono i connettivi del discorso (congiunzione, scelta, negazione: “e”, “o”, “non”; ma anche: aut-aut, implicazione, etc...)
- **Si può dimostrare che l’insieme degli operatori AND, OR, NOT è un insieme funzionalmente completo**
- Quindi ogni proposizione nel calcolo proposizionale può essere espressa mediante gli operatori AND, OR, NOT

Regole di precedenza



- In ordine di priorità dalla maggiore alla minore:

NOT

AND

OR

- Esempi (a,b e c rappresentano proposizioni logiche):

a OR b AND c	equivale a	a OR (b AND c)
a OR NOT b	equivale a	a OR (NOT b)
a OR NOT b AND c	equivale a	a OR ((NOT b) AND c)

Esempi



- Proposizione **A**: **oggi è martedì**
 - Proposizione **B**: **oggi è il 2 novembre**
 - **NOT A** (oggi non è martedì)
 - **NOT B** (oggi non è il 2 novembre)
 - **A AND B** (oggi è martedì 2 novembre)
 - **A OR B** (oggi è martedì oppure è il 2 novembre)
 - **NOT A AND B** (oggi non è martedì ed è il 2 novembre)
 - **A AND NOT B** (oggi è martedì e non è il 2 novembre)
 - **NOT A OR B** (oggi non è martedì oppure è il 2 novembre)
 - **A OR NOT B** (oggi è martedì oppure non è il 2 novembre)
 - **A XOR B** (o oggi è martedì, o è il 2 novembre)
-
- **ATTENZIONE A NON CONFONDERE L'OR (INCLUSIVO) CON XOR (AUT-AUT)**

Tabelle di Verità



- Il valore di verità di una proposizione composta dipende dal valore di verità delle proposizioni atomiche che la compongono
- I connettivi logici in quanto operatori sono funzioni che associano ad ogni valore di verità un valore corrispondente
- In quanto funzioni possono essere definiti in maniera tabellare (Tabelle di Verità)
- Le proposizioni composte sono espressioni logiche dette anche funzioni booleane
- Di conseguenza una proposizione composta (o funzione booleana) è esprimibile (e valutabile) mediante tabelle di verità

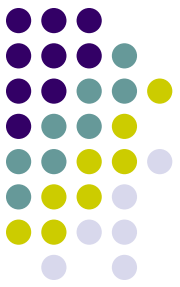
Costruzione di una Tabella di Verità (1/2)



- In generale dunque una tabella di verità è la definizione tabellare di una funzione booleana in K variabili (in questo caso le variabili sono proposizioni atomiche)
- **Per ogni possibile combinazione dei valori delle variabili** deve essere specificato il valore assunto dalla funzione
- Le variabili possono assumere solo due valori (VERO o FALSO)
- Le possibili combinazioni sono 2^k

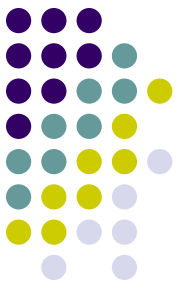
Costruzione di una Tabella di Verità

(2/2)



- La tabella di verità di una funzione booleana in K variabili ha 2^k righe e $K+1$ colonne (una per ogni variabile più una colonna per il valore assunto dalla funzione)
- Le combinazioni sono ordinate in modo tale che sostituendo falso e vero con le cifre binarie 0 e 1 le combinazioni siano ordinate in ordine crescente rispetto al valore numerico ad esse associato dalla rappresentazione posizionale binaria
- In questo modo è possibile:
 - Assicurarsi facilmente di non aver «dimenticato» alcune combinazioni
 - Confrontare tra loro le tabelle di verità perché le combinazioni saranno espresse sempre nello stesso ordine

Esempio



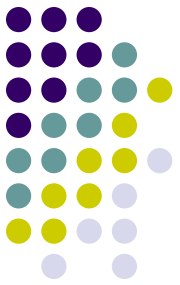
- Utilizziamo il connettore logico AND come operatore BINARIO
- In questo caso AND prevede due operandi (variabili)
- Ad ogni possibile combinazione dei valori delle due variabili deve essere specificato il valore assunto dall'operatore
- Poiché gli operatori possono assumere due valori (VERO o FALSO) le possibili combinazioni sono 4
- Pertanto la tabella di verità corrispondente è costituita da 4 righe e 3 colonne

Operatore AND binario (congiunzione logica)



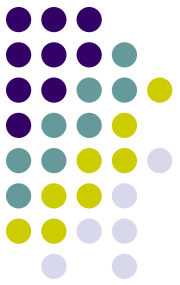
OPERANDI		RISULTATO
A	B	C
F	F	F
F	V	F
V	F	F
V	V	V

Operatore OR binario (disgiunzione logica)



OPERANDI		RISULTATO
A	B	C
F	F	F
F	V	V
V	F	V
V	V	V

Operatore NOT (negazione logica)



OPERANDO	RISULTATO
A	NOT A
F	V
V	F

Esempio: XOR



- Vogliamo ottenere **l'espressione booleana** dell'OR "esclusivo" (aut-aut)
- **Primo passo:** specificare la tabella di verità

OPERANDI		RISULTATO
A	B	C
F	F	F
F	V	V
V	F	V
V	V	F

Esempio: XOR



- Secondo passo: derivare la funzione booleana dalla tabella

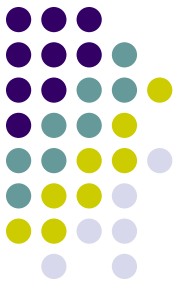
OPERANDI		RISULTATO
A	B	C
F	F	F
F	V	V
V	F	V
V	V	F

Dalla tabella risulta che C è VERO **se e solo se**:

(A è FALSO e B è VERO) oppure (A è VERO e B è FALSO). Quindi:

((NOT A) AND B) OR (A AND (NOT B))

Esercizio: operatori di confronto (1/2)



- Costruire la tabella di verità della seguente espressione logica:

$$Y = ((X \geq 5) \text{ e } (X \leq 10))$$

$X \geq 5$	$X \leq 10$	Y
F	F	F
F	V	F
V	F	F
V	V	V

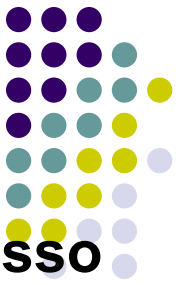
Gli operatori di confronto permettono di creare proposizioni su insiemi non esclusivamente logici (es. su insiemi numerici)

Per quali valori interi di X l'espressione è vera?

$X \in \{5, 6, 7, 8, 9, 10\}$

ovvero quando il numero X appartiene all'intervallo [5; 10].

Proprietà (1/3)



- **Negando due volte un elemento si ottiene l'elemento stesso (Convoluzione)**
- **NOT(NOT a) = a**
- **FALSO è l'elemento neutro della operazione OR**
- **VERO è l'elemento neutro della operazione AND**
- **Quindi :**
- **a OR FALSO = a**
- **a AND VERO = a**

Proprietà (2/3)



commutativa:

$$a \text{ OR } b = b \text{ OR } a$$
$$a \text{ AND } b = b \text{ AND } a$$

associativa:

$$(a \text{ OR } b) \text{ OR } c = a \text{ OR } (b \text{ OR } c)$$
$$(a \text{ AND } b) \text{ AND } c = a \text{ AND } (b \text{ AND } c)$$

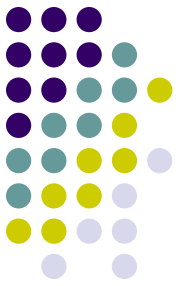
assorbimento:

$$a \text{ OR } (a \text{ AND } b) = a$$
$$a \text{ AND } (a \text{ OR } b) = a$$

idempotenza:

$$a \text{ OR } a = a$$
$$a \text{ AND } a = a$$

Proprietà (3/3)



distributiva:

$$a \text{ AND } (b \text{ OR } c) = (a \text{ AND } b) \text{ OR } (a \text{ AND } c)$$

$$a \text{ OR } (b \text{ AND } c) = (a \text{ OR } b) \text{ AND } (a \text{ OR } c)$$

del minimo e del massimo:

$$a \text{ AND } F = F$$

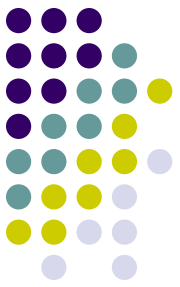
$$a \text{ OR } V = V$$

del complemento:

$$a \text{ AND } (\text{NOT } a) = F$$

$$a \text{ OR } (\text{NOT } a) = V$$

Esempio di dimostrazione di proprietà mediante tabelle di verità



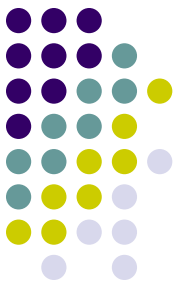
- Idempotenza

$$a \text{ OR } a = a$$

$$a \text{ AND } a = a$$

OPERANDI		RISULTATO
A	A	C
F	F	F
V	V	V

Esempio di dimostrazione di proprietà mediante tabelle di verità



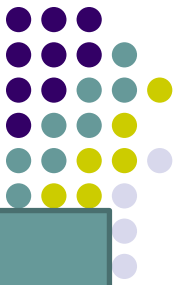
- Del complemento

$$a \text{ AND } (\text{NOT } a) = F$$

$$a \text{ OR } (\text{NOT } a) = V$$

OPERANDI		RISULTATO	RISULTATO
A	NOT A	C	C
F	V	F	V
V	F	F	V

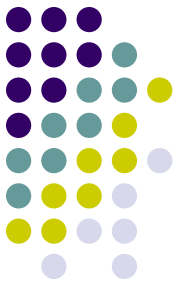
Dualità



- Legge di dualità:

Da qualsiasi *identità* booleana se ne può trarre un'altra per dualità, sostituendo cioè ad ogni operatore e ai valori F e T il rispettivo duale

- La legge di dualità ha come oggetto IDENTITA' booleane, cioè si applica ad «uguaglianze», dice che ad ogni uguaglianza ne corrisponde sempre una duale ottenuta sostituendo l'operatore AND con l'operatore OR, l'operatore OR con AND, il valore di verità F con V e il valore di verità V con F.



Riferimenti

- Dal libro di testo:
 - Teoria Capitolo 1 (parte centrale)
- Dalle dispense del Prof. Iannello: Capitolo 2