1 | Proprietà delle relazioni

Una **relazione binaria** R (con Dominio S) può soddisfare le seguenti proprietà:

- **Riflessiva,** cioè $\langle x, x\rangle \in R$ per ogni $x \in S$, cioè ogni x è in relazione con se stesso
- **Irriflessiva,** cioè $\langle x, x\rangle \notin R$ per ogni $x \in S$, cioè nemmeno un x è in relazione con se stesso
- **Simmetrica,** cioè ogni volta che esite $\langle x, y\rangle \in R$ esiste anche $\langle y, x\rangle \in R$
- **Asimmetrica,** cioè ogni volta che eiste $\langle x, y\rangle \in R$ non esiste mai $\langle y, x\rangle \in R$
- **Antisimmetrica,** cioè che se $\langle x, y\rangle \in R$ e $\langle y, x\rangle \in R$ possiamo dire che $x = y$
- **Transitiva,** cioè se $\langle x, y\rangle \in R$ e $\langle y, z\rangle \in R$ c'è anche $\langle x, z\rangle \in R$

1.1 Chiarimenti sulla proprietà Antisimmetrica

Se x in relazione con y e y in relazione con x allora $x = y$

1.1.1 Esempio

Considero l'insieme degli abitanti dell'Italia e considero la relazione "abita nella stessa città" la relazione non è antisimmetrica: infatti se Maria abita nella stessa città di Carlo e Carlo abita nella stessa città di Maria non segue che Carlo è uguale a Maria

Considero i numeri naturali e considero la relazione "è maggiore o uguale a" La relazione è antisimmetrica perché perché se un numero è maggiore o uguale ad un secondo numero ed il secondo è maggiore uguale del primo allora i due numeri sono uguali

Nella rappresentazione a Grafi si capisce che è antisimmetrica perché ha cicli di lunghezza massima 1 (Solo cappi ammessi)

1.2 Proposizioni derivate dalle proprietà

Dalla *riflessiva*:

- Se R è riflessiva anche R^{-1} (l'inversa) è riflessiva
- R è riflessiva se e solo se $\bar{R}$ è irriflessiva
- Se R e R' sono riflessive anche $R \cup R'$ e $R \cap R'$ sono riflessive

Dalla *simmetrica*:

- R è simmetrica se e solo se $R = R^{-1}$
- se R è simmetrica anche R^{-1} e $\bar{R}$ sono simmetriche
- se R e R' sono simmetriche anche $R \cup R'$ e $R \cap R'$ sono simmetriche

Dall' *antisimmetrica*:

- R è antisimmetrica se e solo se $R \cap R^{-1} \subseteq \varphi S$
- R è antisimmetrica se e solo se $R \cup R^{-1} = \emptyset$

Dalla *transitiva*:

- Se R e R' sono transitive anche $R \cap R'$ è transitiva

2 | Metodi di rappresentazione delle relazioni

2.1 Tabella

Le **relazioni n-arie** (cioè di *arietà pari a n*) possono essere *sempre* rappresentate mediante una **tabella**. La tabella ha n colonne. In particolare se la relazione da rappresentare (che chiamiamo R) è un sottoinsieme del prodotto cartesiano S *todo*

2.2 Matrice

Operazioni su Matrici Booleane

2.3 Grafo bipartito

2.4 Grafo orientato

Con **grafo orientato** (o grafo *diretto* o *digrafo*) intendiamo un metodo di rappresentare una relazione binaria G definitia su un solo insieme V tale che $G \subseteq V \times V$.

Un grafo viene generalmente raffigurato sul piano da punti, che rappresentano i nodi; archi o spigoli sono rappresentati da segmenti o curve che collegano due nodi. In questo caso, il posizionamento dei nodi e la forma degli archi o spigoli è irrilevante, dal momento che a contare sono solo i nodi e le relazioni tra essi. In altri termini, lo stesso grafo può essere disegnato in molti modi diversi senza modificarne le proprietà.

2.4.1 Nodi e archi

Gli elementi di V sono detti **nodi** (o vertici), gli elementi di G sono detti **archi**.

Un arco che va da vi a vj si dice **uscente** da vi ed **entrante** in vj, o più generalmente **incidente** da vi a vj (e quindi vi e vj sono detti **adiacente** tra loro).

Il **numero di archi uscenti** da un nodo è detto il **grado di uscita** del nodo. Il **numero di archi entranti** da un nodo è detto il **grado di entrata** del nodo.

Nodo Sorgente se non ha archi entranti. **Nodo Pozzo** se non ha archi uscenti.

Aggiungere cose sulle matrici

Un nodo è detto **isolato** se non ha archi entranti o uscenti.

2.4.2 Cammini e cicli

Un **cammino** tra due nodi vin e $vfin$ è una sequenza finita di nodi $\langle vin, v2, v3, ..., vfin\rangle$ dove ciascun nodo è collegato al successivo da un arco uscente dal primo ed entrante nel secondo.

Un **semicammino** tra due nodi vin e $vfin$ è una sequenza finita di nodi $\langle vin, v2, v3, ..., vfin\rangle$ dove ciascun nodo è collegato al successivo da un arco di direzione arbitraria.

Si dice **lunghezza** di un cammino il numero di archi che lo compongono. La lunghezza di un cammino è uguale al numero di nodi che lo compongono meno 1.

Un grafo si dice **connesso** se dati due nodi qualunque **esiste sempre** un **semicammino** che li connette. Un grafo si dice **fortemente connesso** se dati due nodi qualunque **esiste sempre** un **cammino** che li connette.

Un **ciclo** intorno ad un nodo v di un grafo è un cammino in cui $v = vin = vfin$ Un **semiciclo** intorno ad un nodo v di un grafo è un semicammino in cui $v = vin = vfin$.

Se il **ciclo di lunghezza 1** esso viene chiamato **cappio**, che è un arco che esce dal nodo per poi subito rientrare.

Il minimo numero di archi che compongono un cammino tra due nodi vin e $vfin$ è detta distanza tra i due nodi.

2.4.3 Grafo etichettato

Un **grafo etichettato** è una funzione che **associa ad ogni arco** del grafo **un'etichetta**, cioè una sorta di nome.

2.4.4 Grafo completo

Si ha un grafo completo quando ogni coppia di vertici del grafo ha un arco che li unisce.

2.4.5 Grafi e proprietà delle relazioni

G è una relazione binaria.

- Se G è **riflessiva** allora il grafo di G avrà un **cappio intorno ad ogni nodo**. Similmente se G è **irriflessiva** non c'è **mai un cappio nel grafo** associato.
- Se G è **simmetrica** allora ogni volta che nel grafo c'è un arco tra due nodi c'è anche quello che va in direzione opposta. Similmente se è **asimmetrica** se un arco congiunge due nodi non c'è mai il suo opposto.
- Se G è **transitiva** allora ogni volta che nel grafo associato abbiamo una situazione $v1- > v2, v2- > v3$ (si potrebbe dire che ci sono archi di fila tra tre nodi) ci sarà per forza anche un arco che collega v1 a v3 (che chiude il triangolo v1,v2,v3).

3 | Relazioni Particolari

3.1 DAG

Chiamiamo **DAG** (o “Grafo Diretto Aciclico”) un **grafo diretto senza cicli**.

Un DAG è riconoscibile perchè nella sua rappresentazione **gli archi vanno tutti verso una sola direzione** e non essendoci cicli non possono andare nella direzione opposta.

Ad esempio tutti gli archi uscenti dai vari nodi puntano ad elementi disposti graficamente piu’ in basso.

L’esempio più comune di DAG è dato dalla particolare relazione che chiamiamo **alberi**.

3.2 Alberi

Un **albero** è un DAG connesso che ha un solo node sorgente (che per comodità chiamiamo **radice**). Tutti i nodi dell’albero, ad eccezione della radice, hanno **uno ed un solo arco entrante**. **La radice non ha archi entranti ma solo archi uscenti**. I nodi di un albero possono avere **da 0 a N archi uscenti**. I nodi con 0 archi uscenti sono detti **foglie** dell’albero.

Un particolare tipo di albero è l’**albero binario**, particolare relazione per cui **ogni nodo ha al massimo 2 nodi figli** che per comodità denominiamo **figlio destro** e **figlio sinistro**

Graficamente, quando disegnamo un albero, è possibile evitare di disegnare le frecce che connettono i nodi: infatti riconoscere la radice è semplice, basta trovare il nodo che non ha archi entranti. Una volta trovata la radice si sa che, essendo l’albero un DAG, tutti gli archi che “coinvolgono” un nodo vanno nella stessa direzione e mai nel senso opposto degli altri. Non ci sono quindi ambiguità. È generalmente consigliabile mettere le frecce nel disegnare gli alberi.

Preso un nodo dell’albero possiamo determinare la sua **profondità** all’interno dell’albero: il numero di archi necessari per andare dalla radice dell'albero al nodo stesso.

Si dice invece **altezza** di un albero il valore di profondità massima raggiungibile dai nodi degli alberi, cioè la distanza massima che c'è da una foglia, tra tutte le foglie, e la radice dell'albero.

3.3 Relazioni di Equivalenza

Diciamo che R è una **relazione di equivalenza** su un insieme S se e solo se R è **binaria su** S**, riflessiva, simmetrica e transitiva**.

3.3.1 Esempio

Un esempio di relazione di equivalenza è data dalla seguente.

$R = \langle a, b \rangle \in Z | a^2 = b^2$

Essa infatti è:

- *binaria* (ci sono due "termini" coinvolti nella relazione, a e b)
- *riflessiva* ($n^2 = n^2$ per ogni $n \in Z$)
- *simmetrica* (se $n^2 = m^2$ anche $m^2 = n^2$ per ogni $m, n \in Z$)
- *transitiva* (se $n^2 = o^2$ e $m^2 = o^2$ allora $n^2 = o^2$ per ogni $m, n, o \in Z$)

3.3.2 Classe di Equivalenza

Data una relazione di equivalenza R su un insieme S, la **classse di equivalenza** di un elemento $x \in S$ è definita come

$[x] = y | \langle x, y \rangle \in R$

Cioè la **classe di equivalenza** di un elemento $x \in R$ è **l'insieme degli elementi di** R **con cui** x **è in relazione**.

Se R è una relazione di equivalenza su S allora le classi di equivalenza generate da R partizionano S. (Vedere i capitoli precedenti per la definizione di *classe di equivalenza*).

3.4 Relazioni Composte

Date due relazioni $R1$ e $R2$ con $R1$ definita su $S \times T$ e $R2$ definita su $T \times Q$ è possibile definire una nuova relazione che chiamiamo di composizione tra $R1$ e $R2$ su $S \times Q$ tale che $\langle a, c \rangle \in R2 \circ R1$ se e solo se esiste un $b \in T$ tale che $\langle a, b \rangle \in R1$ e $\langle b, c \rangle \in R2$.

$R2 \circ R1$ è detta composizione di $R1$ e $R2$.

Aggiungere considerazioni sul dominio

3.4.1 Proprietà della composizione

Scrivere $R2 \circ R1$ è diverso dallo scrivere $R1 \circ R2$, **la composizione non è un'operazione commutativa**.

La composizione è un'**operazione è associativa**.

Prese due relazioni $R1$ e $R2$ con $R1$ definita su $S \times T$ e $R2$ definita su $T \times Q$, si ha che $(R1 \circ R2)^{-1} = R1^{-1} \circ R2^{-1}$.

4 | Strutture Relazionali

Definiamo una *struttura relazionale* (SR) come una n-upla ($\langle ... \rangle$) in cui il primo elemento è un insieme non vuoto S, chiamato dominio o universo di SR, e i rimanenti elementi sono relazioni (che possono avere arietà varia) definite su S.

I casi da noi studiati sono solitamente caratterizzati dall'avere un insieme S (richiesto per definizione) ed una sola relazione associata, generalmente binaria.

4.0.1 Esempio

Insieme Universo

U = Insieme N (dei numeri naturali)

Relazioni

$R1 = n2 = m$ = relazione che associa ad ogni numero il suo quadrato

$R2 = n + m = o$ = relazione che associa a due numeri la loro somma

$SR = \langle U, R1, R2 \rangle$

4.1 Strutture Relazionali Notevoli

4.1.1 Preordine

Un **preordine** è una struttura relazionale data da una coppia $\langle S, R \rangle$ in cui S è un insieme ed R è una relazione **binaria, riflessiva** e **transitiva** su S

4.1.2 Quasi-ordine

Un **quasi-ordine** è una struttura relazionale data da una coppia $\langle S, R \rangle$ in cui S è un insieme ed R è una relazione **binaria, irriflessiva** e **transitiva** su S.

4.1.3 Poset

Un **poset** (chiamato anche *ordine parziale* o *semiordinamento*) è una struttura relazionale data da una coppia $\langle S, R \rangle$ in cui S è un insieme e R è una relazione **binaria, riflessiva, antisimmetrica** e **transitiva**.

4.1.4 Ordinamento

Particolare semiordinamento che esiste se e solo se per ogni $x, y \in S$ una ed una sola delle seguenti condizioni è soddisfatta:

- $x = y$
- $\langle x, y \rangle \in R$
- $\langle y, x \rangle \in R$

Detto semplicemente, in un ordinamento **presi due elementi** dell'insieme di partenza (o insieme universo) **questi sono confrontabili**: posso dire se sono uguali oppure chi è il maggiore e il minore tra i due.

L'ordinamento prende anche il nome di **ordine totale**.

4.1.5 Buon Ordinamento

todo

4.1.6 Ordini totali vs ordini parziali

Possiamo facilmente distingurere tra un poset (ordine parziale) e un ordinamento (ordine totale) osservando la rappresentazione del grafo associato ad entrambe le strutture. Nel caso di un ordine totale tutti gli elementi della struttura relazionale sono collegati e confrontabili tra loro, cosa che non è vera per gli ordini parziali, in cui alcuni elementi sono completamente isolati dagli altri.

4.2 Reticoli

I **reticoli** sono poset in cui per ogni coppia di $\langle x, y \rangle \in S$ esistono un mimino maggiorante e un massimo minorante.

5 | Poset

5.1 Definizione

Un **poset** (chiamato anche *ordine parziale* o *semiordinamento*) è una struttura relazionale data da una coppia $\langle S, R \rangle$ in cui S è un insieme e R è una relazione **binaria**, **riflessiva**, **antisimmetrica** e **transitiva**.

5.2 Rappresentazione grafica

Rappresentato il grafo di un poset ci si accorge di alcune caratteristiche distintive:

- il grafo di un poset non ha cicli di lunghezza maggiore di 1 (antisimmetria della relazione)
- più precisamente, il grafo ha solo cicli di lunghezza 1 (antisimmetria della relazione e riflessività)

La rappresentazione classica ha però degli elementi superflui, non necessariamente utili alla comprensione del poset e che la rendono pesante da disegnare e comprendere.

Per risolvere tale problema vengono in aiuto i **Diagrammi di Hasse**, particolari rappresentazioni di Poset basate su delle convenzioni comuni.

In particolare:

- si eliminano i cappi dalla rappresentazione, si sa a prescindere che esistono visto che si sta lavorando con un poset (riflessivo)
- si eliminano gli archi della transitività, come sopra, sono superflui, se $a- > b$ e $b- > c$ non rappresentiamo l'arco $a- > c$ anche se per definizione dovrebbe esserci. Tale operazione è chiamata riduzione transitiva.
- eliminiamo completamente le frecce che indicano la direzione degli archi rimasti. Se dobbiamo rappresentare $\langle a, b \rangle$ metteremo a più in basso di b, in modo che tutte le

eventuali frecce puntassero verso l'alto.

In conclusione:

- gli unici archi che rimangono sono quelli che indicano le coperture tra gli elementi

5.2.1 Copertura

Il concetto di copertura torna particolarmente utile quando si disegnano i diagrammi di Hasse, ma non per questo è limitato solo ad essi.

Diamo per prima cosa una definizione generica.

Presi tre elementi diversi x, y, z appartenenti alla relazione R. Si dice che y è una copertura di x se $\langle x, y\rangle \in R$ (cioè se x è in relazione con y) e allo stesso tempo non esiste che $\langle x, z\rangle \in R$ e non esiste che $\langle z, y\rangle \in R$.

A parole potremmo dire che x è copertura di y se non esiste z che si mette in mezzo tra il confronto di x e y.

A partire da un grafo, per ottenere tutte le possibili coperture basta fare l'operazione di riduzione transitiva.

Esempio

$A = 1, 2, 3, 4, 5, 6, 7, 8$

R è espressa dalla proprietà $\langle x, y\rangle$ tale che x è divisibile per y

Quindi 4 è copertura di 8.

Come anche 2 è copertura di 4.

2 però non è copertura di 8, perchè anche se 8 è divisibile per 2, esiste comunque 4 (che sta in mezzo tra il 2 e l'8) che è divisore (e copertura) di 8.

5.3 Elementi estremali

In un poset è possibile individuare alcuni particolari elementi che godono di determinate proprietà chiamati elementi estremali.

5.3.1 Minimale e Massimale

Dato un poset S, s_min è detto elemento minimale di S se per ogni elemento $s' \in S$ $s\_min < s'$.

Dato un poset S, s_max è detto elemento massimale di S se per ogni elemento $s' \in S$ $s\_max > s'$.

Dato un poset S, esiste almeno un elemento massimale e uno minimale.

5.3.2 Minorante e Maggiorante

Dato un poset S e un sottoinsieme $A \subseteq S$, è detto **minorante** di A un elemento di S tale che per ogni $a \in A$ $s \leq x$. Possono esistere più minoranti.

Dato un poset S e un sottoinsieme $A \subseteq S$, è detto **maggiorante** di A un elemento di S tale che per ogni $a \in A$ $s \geq x$. Possono esistere più maggioranti.

5.3.3 Massimo Minorante e Minimo Maggiorante

Dato un poset S e un sottoinsieme $A \subseteq S$, è detto massimo minorante (greatest lower bound) di A un elemento di S che è minorante di A ed è maggiore o uguale di tutti gli altri minoranti di A.

Dato un poset S e un sottoinsieme $A \subseteq S$, è detto minimo maggiorante (least upper bound) di A un elemento di S che è maggiorante di A ed è minore o uguale di tutti gli altri maggioranti di A.

Dato un poset S, ogni sottoinsime $A \subseteq S$ ha al massimo un minimo maggiorante e un massimo minorante.

5.3.4 $\leq$-massimo e $\leq$-minimo

È detto $\leq$-massimo l'elemento $\geq$ di tutti gli altri elementi di un poset S.

È detto $\leq$-minimo l'elemento $\leq$ di tutti gli altri elementi di un poset S.

Dato un poset s, esiste al piu un $\leq$-minimo e un $\leq$-massimo.

6 | Reticoli

6.1 Definizione

I reticoli sono poset in cui per ogni coppia di $\langle x, y \rangle \in S$ esistono un mimino maggiorante (che chiamiamo *join* e che indichiamo con il simbolo $\sqcup$) e un massimo minorante (che chiamiamo *meet* e che indichiamo con il simbolo $\sqcap$).

6.2 Proprietà e proposizioni dei reticoli

6.2.1 Complemento di un elemento

Preso un reticolo L, un elemento $a' \in L$ è detto complemento di $a \in L$ se a e a' hanno lo stesso meet e join. (Il libro dice se hanno lo stesso minimo minorante e massimo maggiorante)

6.3 Reticoli particolari

6.3.1 Reticolo Completo

Si chiama reticolo completo un reticolo L in cui ogni sottoinsieme A di L ha un minimo maggiorante (non vuoto) e un massimo minorante (non vuoto).

6.3.2 Reticolo Limitato

Si chiama reticolo limitato un reticolo L in cui esiste un elemento massimo ed uno minimo.

6.3.3 Reticolo distributivo

Si chiama reticolo distributivo un reticolo L in cui per ogni terna di elementi $x, y, z \in L$ valgono le seguenti proprietà, che chiamiamo distributive:

- $x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z)$
- $x \sqcup (y \sqcap z) = (x \sqcup y) \sqcap (x \sqcup z)$

7 | Induzione

L'induzione è un metodo di ragionamento che è utile per:

- Verificare proprietà di strutture di dati
- Definire strutture dati
- Definire procedimenti di calcolo su strutture dati attraverso la ricorsione

7.1 Definizione

7.1.1 Principio di Induzione Matematica

Sia $A(n)$ una asserzione (proprietà da verificare) per ogni elemento dell'insieme dei numeri naturali N. Supponendo che:

- $A(n)$ è vera
- per ogni $k \in N$, se A(k) è vera, allora è vera anche $A(k+1)$

Allora per ogni $n \in N$, $A(n)$ è vera.

7.1.2 Principio di Induzione Matematica Completo

Sia $A(n)$ una asserzione (proprietà da verificare) per ogni elemento dell'insieme dei numeri naturali N. Supponendo che:

- $A(n)$ è vera
- per ogni $m \in N$, se $A(k)$ è vera per ogni k, con $0 < k < m$, ne segue che è vera $A(m)$

Allora per ogni $n \in N$ e $n > 0$, $A(n)$ è vera.

7.2 Esempi di Dimostrazione per Induzione

7.3 Dichiarazione di insiemi tramite l'induzione

8 | Algebra Booleana

8.1 Definizione

Si definisce **algebra booleana** un **reticolo** $\langle B, R\rangle$ caratterizzato dall'essere **distributivo, limitato** e **complementato**.

Più precisamente, si parla di algebra di Boole in riferimento ad un insieme B sul quale sono definite due operazioni binarie $\sqcap$ e $\sqcup$, un'operazione unaria ('), e $\underline{1}$ e $\underline{0}$ due costanti, in modo da ottenere la sestupla ordinata(?) $\langle B, \sqcap, \sqcup, ', \underline{0}, \underline{1}\rangle$, che costituisce un reticolo distributivo, limitato e complementato.

Chiarimento

Generalmente l'operazione $\sqcap$ è chiamata **AND** e si indica anche con $*$, l'operazione $\sqcup$ è chiamata **OR** e si indica anche con $+$ e l'operazione $'$ è chiamata **NOT** e si indica anche con ! o con $\neg$ o anche con $-$.

8.2 Enunciato

E' detto enunciato booleano una proposizione della quale si può sempre dire se è vera o falsa (non lascia cioè spazio ad ambiguità).

Nell'algebra di Boole un enunciato è ottenuto dalla combinazione di **termini** e **operatori.**

8.3 Termini

I termini di un enunciato sono essi stessi degli enunciati, ovviamente meno complessi dell'enunciato che sto costruendo. Preso un termine posso infatti dire con certezza se è vero o falso.

Tra tutti i termini riconosciamo dei termini elementari che possono essere o delle costanti ($\underline{0}$ e $\underline{1}$) o delle variabili, generalmente indicate con le lettere minuscole a, b, c, ..., che come suggerisce il nome variano a seconda dei casi.

La presenza di variabili non lascia comunque spazio ad ambiguità perché, sostituita una variabile con $\underline{0}$ o con $\underline{1}$, posso sempre dire se l'espressione che contiene le varibili è vera o falsa.

8.4 Proprietà Algebra di Boole

Commutativa $a + b = b + a \qquad a * b = b * a$

Associativa $a + (b + c) = (a + b) + c \qquad a * (b * c) = (a * b) * c$

Assorbimento $a + (a * b) = a \qquad a * (a + b) = a$

Distributiva $a * (b + c) = (a * b) + (a * c) \qquad a + (b * c) = (a + b) * (a + c)$

Idempotenza $a + a = a \qquad a * a = a$

Esistenza di minimo e massimo $a * 0 = 0 \qquad a + 1 = 1$

Esistenza del complemento $a*!a = 0 \qquad a+!a = 1$

8.5 Extra

La **più piccola algebra di Boole** esistente è quella definita sull'insime {0,1}.

È possibile definire un'algebra di Boole su ogni insieme che ha come numero di elementi una potenza di due.

È detto enunciato duale D' di un enunciato D, l'enunciato ottenuto scambiando $\sqcap$ con $\sqcup$ e $\underline{1}$ con $\underline{0}$. Se è dimostrabile D anche il suo duale D' sarà dimostrabile.

9 | Logica

Chiamata anche *sistema formale*, una logica Lo è definita da:

- Alf, **alfabeto** di Lo: un insieme non vuoto, finito o numerabile, di simboli. Combinando più simboli si ottiene un'espressione di Lo.
- F, insieme delle formule ben formate: sottoinsieme di espressioni di Lo che come suggerisce il nome contiene le FBF di Lo.
- $Alfx$, insieme degli assiomi di Lo.
- R, insieme finito di relazioni $R1, R2, ..., Rn$: queste relazioni sono definite tra le formule di Lo e sono dette **regole di inferenza**.

9.1 Linguaggio

Presa una logica A, chiamiamo L il linguaggio della logica, costruito induttivamente, e che equivale all'insieme delle FBF di A.

Visto che l'insieme delle FBF è costruito induttivamente esso rende il linguaggio L della logica A un **linguaggio composizionale**: combinando infatti le FBF esistenti se ne possono ottenere nuove.

9.2 Formule ben formate

Per definire l'insieme delle formule ben formate F di un linguaggio L è necessario utilizzare l'induzione.

Per prima cosa definiamo un insieme non vuoto che chiamiamo $Atom$ (insieme degli atomi) ed un insieme non vuoto Op di operatori unari (che indichiamo con $?$) e binari (che indichiamo con o). Definiamo $L = Atom \cup Op \cup (,)$

Allora F è definito induttivamente:

- Se A è un elemento di $Atom$ allora $A \in F$
- Se $A \in F$ e ? è un operatore unario allora $(?A) \in F$
- Se $A \in F$, $B \in F$ e o è un operatore binario allora $(AoB) \in F$

9.2.1 Unicità della scomposizione

Per ogni FBF vale una (ed una sola) delle seguenti proprietà:

- $A \in Atom$ (formula atomica)
- Esiste un unico operatore unario $?$ e un'unica formula B tal che A ha la forma $(?B)$
- Esiste un unico operatore binario o e due formule uniche B e C tali che A ha la forma (BoC)

9.3 Interpretazione del linguaggio

Preso un qualsiasi linguaggio logico (ma anche un linguaggio come l'italiano o un linguaggio di programmazione) e costruita una proposizione combinando elementi del linguaggio, è necessario interpretarla cioè andare a stabilirne il significato. Non è abbastanza studiare la frase o la proposizione guardando soltanto i simboli che la compongono (analisi sintattica) ma è necessario anche comprenderne a pieno il senso (analisi semantica).

Attraverso l'inferenza -> ragionamento per arrivare ad un risultato, una sorta di dimostrazione.

Nelle regole di inferenza c'è una premessa e una conclusione.

Esempio

Prendiamo ad esempio le due frasi:

- "Tutti i cani sono animali"
- "Tutti gli animali sono cani"

Anche se le parole usate per comporre le due frasi sono le stesse il loro significato è completamente diverso.

9.3.1 Interpretazione nella logica formale

Nel caso di un sistema logico formale associare ad una frase una semantica (un senso) significa determinare il contesto di interpretazione della frase, attraverso un operazione chiamata funzione di interpretazione. Per fare questa operazione viene fornito un insieme B (insieme dei valori di verità), caratterizzato dall'avere almeno due elementi ed un insieme T (insieme designato a rappresentare il vero). La funzione di interpretazione restituirà un valore di B.

Generalmente B ha due elementi (t, f) ma è anche possibile che B abbia un qualsiasi numero finito o non di elementi.

Soddisfacibilità

Nel caso sia necessario è possibile studiare il significato di una formula nel contesto di una struttura o realtà matematica, attraverso l'utilizzo della relazione di *soddisfacibilità* che si indica con il simbolo $\models$. $\models$ è sottoinsieme di $M \times F$ dove M è una struttura e $M \models A$ (M modella A).

9.4 Modelli

Preso un insieme F di formule ben formate, che sono utilizzate per rappresentare la realtà, chiamiamo **modelli** di F le formule di F che sono verificate. L'insieme F prende anche il nome di insieme degli *assiomi associati*.

9.4.1 Rappresentazione della realtà

Per rappresentare una realtà usando soltanto una logica A dobbiamo quindi servirci di vari modelli che (passatemi il gioco di parole) **modellano la realtà**, regole logiche che sono rispettate e verificate all'interno della realtà di riferimento.

Esempio

Realtà da modellare: Pianeta Terra

La Luna è un satellite della Terra = Vero, è un modello

Sulla terra c'è acqua = Vero, è un modello

Sulla terra non esistono animali = Falso, non è un modello

Descrizione della terra attraverso i modelli: La Terra ha come satellite la Luna e sulla Terra c'è acqua.

9.5 Albero sintattico

Un albero sintattico T è un albero binario con nodi etichettati da simboli di L che indica la precisa ed unica scomposizione di una FBF X.

- Se X è una formula atomica, l'albero binario X ha solo un nodo etichettato con X
- Se $X = AoB$, X è rappresentato dall'albero binario che ha la radice etichettata con o il cui figlio sinistro è la rappresentazione di A e quello destro quella di B.
- Se $X = ?A$, X è la rappresentazione dell'albero binario che ha la radce etichettata con $?$ e il suo unico figlio è la rappresentazione di A

10 | Sintassi Logica Proposizionale

La logica proposizionale si occupa di investigare il valore di verità di proposizioni (ben) formate da proposizioni elementari, dette anche atomiche. Più precisamente, lo scopo della logica proposizionale (semantica) è quello di assegnare un valore di verità a proposizioni atomiche (valutazione di verità, tecnicamente) e di capire quindi i valori di verità assunti dalle proposizioni costruite a partire da quelle atomiche secondo precise regole sintattiche.

La sintassi ci permette di capire come scrivere le formule proposizionali e come trattarle.

10.1 Alfabeto proposizionale

Un alfabeto $\sum$ (chiamato anche L se chiaro nel contesto) è costituito da:

- Connettivi proposizionali unari ($\neg$) e binari ($\wedge, \vee, \implies, \iff$)
- Costanti proposizionali $\top$ e $\bot$ (per detonatare Vero e Falso)
- Un insieme non vuoto di simboli proposizionali $P = A, B, C, ...$
- I simboli separatori '(' e ')'

Le costanti e i simboli proposizionali sono chiamati **atomi**, le loro negazioni **atomi negati**. Entrambi sono detti *letterali*.

10.2 Formule proposizionali

L'insieme $PROP$ delle formule ben formate (o formule del linguaggio proposizionale L) è definito per induzione in questo modo:

- Le costanti e i simboli proposizionali sono formule
- Se Q è una formula anche $\neg Q$ è una formula
- Se o è un connettivo logico binario e se Q e R sono due formule, (AoB) è una formula.

Una formula del linguaggio proposizionale è anche chiamata proposizione o anche enunciato proposizionale.

10.2.1 Sottoformule

Data una formula A, è detta sottoformula di A ogni B che appare come componente di A stessa.

Sia A una formula, l'insieme delle sottoformule è definito in questo modo:

- Se A è una costante o un simbolo proposizionale allora A è sottoformula di se stessa.
- Se A è una formula $(\neg Q)$, le sottoformule di A sono A stessa e le sottoformule di Q. $\neg$ è il connettivo logico principale e Q è sottoformula immediata di A.
- Se A è una formula (BoC), dove o è un connettivo binario, le sottoformule di A sono A stessa e le sottoformule di B e C. o è il connettivo principale e B e C sono sottoformule immediate di A.

10.2.2 Esempi di formule

- $\neg A$
- $A \vee B$
- $(\neg A) \wedge (A \vee (\neg(B \implies C)))$

10.3 Parentesi e precedenza tra gli operatori

Se seguissimo alla lettera la definizione di formula proposizionale ci ritroveremmo con formule che hanno tantissime parentesi, perché non è stata ancora specificata una precedenza tra gli operatori.

In particolare nella logica proposizionale gli operatori seguono il seguente ordine di precedenza:

$\neg, \vee, \wedge, \implies, \iff$

Ciò significa che se mancano le parentesi una formula ben formata va parentesizzata prioritizzando le sottoformule i cui connettivi principali hanno la precedenza più alta. A parità di precedenza si associa a destra.

10.3.1 Esempi di priorità

- $\neg A \vee B \wedge C = ((\neg A) \vee B) \wedge C$
- $A \vee B \vee C = A \vee (B \vee C)$

11 | Semantica Logica Proposizionale

La semantica ci permete di dare un senso alle formule proposizionali.

11.1 Sistema di valutazione

Il sistema di valutazione S della logica proposizionale è così definito:

- $B = 0, 1$, il dominio della valutazione logica
- $T = 1$, i valori che assumono il valore di *vero*
- $Op = operatori$, gli operatori utilizzabili

11.2 Valutazione Booleana

Una formula logica proposizionale A è soddisfatta da una valutazione boolena Iv se $Iv(A) = 1$. Cioè se per almeno un caso (o combinazione di veri e falsi) risulta vera.

Una formula logica proposizionale è una tautologia se è soddisfatta per ogni valutazione boolena Iv. Cioè è vera in ogni caso (e quindi indipendentemene dai valori di verità degli enunciati che la compongono).

Una formula logica proposizionale è una contraddizione se non è mai soddisfatta per ogni valutazione boolena Iv. Cioè è falsa in ogni caso (e quindi indipendentemene dai valori di verità degli enunciati che la compongono).

11.3 Operatore negazione

L'operatore $\neg$ è il più semplice tra tutti. $\neg$, come sappiamo dalla sintassi, è un operatore unario. Preso un valore ne restituisce l'opposto logico.

Eccone la tabella di verità:

A	$\neg A$
0	1
1	0

11.4 Operatore AND

L'operatore $\wedge$ è un operatore binario. Detto anche di congiunzione, fa un operazione che, in pratica, equivale al prodotto tra i valori di verità di due proposizioni.

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

Letteralmente $\wedge$ ritorna 1 quando entrambi i "parametri" sono 1.

11.5 Operatore OR

L'operatore $\vee$ è un operatore binario. Detto anche di disgiunzione, fa un operazione che, in pratica, è molto simile alla somma tra i valori di verità di due proposizioni.

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

Letteralmente $\vee$ ritorna 1 quando almeno uno dei due parametri è a 1.

11.6 Operatore Implica

L'operatore $\implies$ è un operatore binario è detto anche di implicazione. A è chiamata premessa e B è chiamata conseguenza.

A	B	$A \implies B$
0	0	1
0	1	1
1	0	0
1	1	1

$A \implies B$ è vero se B è vero, o se sia A che B sono false.

11.7 Operatore co-implica

L'operatore $\iff$ è un operatore binario è detto anche di co-implicazione. Esso può essere scritto come $(A \implies B) \wedge (B \implies A)$.

A	B	$A \iff B$
0	0	1
0	1	0
1	0	0
1	1	1

$A \iff B$ è vero se sia A che B sono entrambe vere o entrambe false, cioè se i loro valori coincidono.

11.8 Equivalenza logica

Diciamo che due proposizioni A e B sono logicamente equivalenti se sono uguali per ogni valutazione booleana (a parità di input, danno lo stesso output).

11.8.1 Idempotenza

$A \wedge A = A$

$A \vee A = A$

11.8.2 Associatività

11.8.3 Commutatività

11.8.4 Distribuitivià

11.8.5 Assorbimento

11.8.6 Doppia negazione

11.8.7 Leggi di De Morgan

11.8.8 Terzo escluso

11.8.9 Contrapposizione

11.8.10 Contraddizione

11.9 Modelli Proposizionali

11.9.1 Definzione

Un modello è un interpretazione di una formuala proposizionale.

11.9.2 Definizione ricorsiva

Sia M un insieme di simboli proposizionali, definiamo $\models \subseteq (M \times L)$ in modo ricorsivo in questo modo:

- $M \models A$ se e solo se $A \in M$
- $M \models Vero$ e $M \not\models Falso$
- $M \models \not A$ se e solo se non $(M \models A)$ se e solo se $M \not\models A$
- $M \models A \wedge B$ se e solo se $M \models A$ e $M \models B$
- $M \models A \vee B$ se e solo se $M \models A$ o $M \models B$
- $M \models A \implies B$ se e solo se $M \not\models A$ o $M \models B$
- $M \models A \iff B$ se e solo se $M \models A$ e $M \models B$ oppure $M \not\models A$ e $M \not\models B$

11.9.3 Valutazione booleana e Modelli

Per dare un significato più concreto a questa definizione possiamo dire che tutti i simboli che appartengono al sottoinsieme M sono quelli che hanno valutazione booleana 1, mentre quelli che non appartengono hanno valutazione booleana 0.

In simboli $p \in M$ se e solo se $V(p) = 1$

In particolare

Il passo base della definizione ricorsiva di $\models$ coincide con la definizione di assegnazione (valutazione) booleana V.

11.9.4 Definizioni relative ai modelli

Sia A una formula, se $M \models A$ diciamo che M è un modello di A, cioè che M rende vera A.

Se M rende vere tutte le formule di un insieme T ($M \models A$ per ogni formula A in T) diciamo che M è un modello per T e indichiamo questo con $M \models T$.

Se $M \models A$ per qualche M allora diciamo che A è soddisfacibile.

Se per nessun insieme di simboli proposizionali M, è verificato che $M \models A$ allora diciamo che A è insoddisfacibile.

11.10 Decidibilità logica proposizionale

La logica proposizionale è decidibile (posso sempre verificare il significato di una formula). Esiste infatti una procedura effettiva che stabilisce la validità o no di una formula, o se questa ad esempio è una tautologia.

In particolare il verificare se una proposizione è tautologica o meno è l'operazione di decibilità principale che si svolge nel calcolo proposizonale.

12 | Sintassi logica predicativa

12.1 Logica Predicativa

Con la logica proposizionale non tutti i concetti e le argomentazioni possono essere espressi, in quanto la validità dell'argomento dipende direttamente dalle relazioni che sussistono tra i *costituenti* delle proposizioni.

Grazie alla logica predicativa possiamo:

- Riferirci a concetti e a individui in particolare
- Fare affermazioni particolari o universali, senza per forza far riferimento a individui specifici.

La logica predicativa prende anche il nome di linguaggio di primo ordine.

Concetti esclusivi alla logica di primo ordine sono, ad esempio, i quantificatori (tutti, alcuno, uno solo).

12.2 Linguaggio logico predicativo

Un linguaggio logico predicativo è costruito sui seguenti simboli logici (immutabili):

- Connettivi proposizionali unari ($\neg$) e binari ($\wedge, \vee, \implies, \iff$)
- Costanti proposizionali $\top$ e $\perp$ (per denotare Vero e Falso)
- Il simbolo di uguaglianza (non sempre presente) $=$
- I simboli separatori '(', ')' e ','
- Un'infinità numerabile di simboli di variabile individuale ($x_1, x_2, x_3, ...$)
- Il simbolo di quantificatore universale $\forall$
- Il simbolo di quantificatore esistenziale $\exists$

e sui seguenti parametri (esclusivi al linguaggio logico predicativo):

- Un insieme finito/numerabile di parametri e simboli di predicato, ognuno dei quali ha associato un intero positivo n detto arità
- Un insieme finito/numerabile di simboli di funzione, ognuno dei quali ha associato un intero positivo n detto arità
- Un insieme finito/numerabile di simboli di costante

Volendo è possibile definire connettivi e componenti al linguaggio logico appena presentato, o addrittura rimuoverne alcuni (perché ad esempio $\exists = \neg\forall\neg$).

12.2.1 Esempi di linguaggi logici predicativi

Linguaggio della teoria degli insiemi

Uguaglianza: è presente (due insiemi possono essere uguali)

Simboli di predicato: $\in$

Simboli di costante: $\emptyset$

Simboli di funzione: nessuno

Linguaggio della teoria dei numeri

Uguaglianza: è presente (due numeri possono essere uguali)

Simboli di predicato: $<$ e $>$

Simboli di costante: 0

Simboli di funzione: s (funzione unaria per trovare il successivo di un numero) e + e - (simboli binari per addizione e sottrazione)

12.3 Espressioni Legali del Linguaggio

Con espressioni legali del linguaggio si intendono tutte le formule di un linguaggio L. Il concetto di espressione legale è strettamente collegato (se non uguale) a quello di formula ben formata.

Per prima cosa definiamo i termini e le formule atomiche.

12.3.1 Termini

L'insieme $TERM$ dei termini di L è l'insieme induttivo definito in questo modo:

- Ogni simbolo di costante e di variabile è un termine
- Se $t_1, \ldots, t_n$ sono termini ed f è un simbolo di funzione n-aria, allora $f(t_1, \ldots, t_n)$ è un termine.

12.3.2 Atomi

L'insieme $ATOM$ delle formule atomiche è definito in questo modo:

- $\top$ e $\bot$ sono atomi
- se t_1 e t_2 sono atomi allora $t_1 = t_2$ è un atomo
- se $t_1, \ldots, t_n$ sono termini e P è un simbolo di predicato a n argomenti, allora $P(t_1, \ldots, t_n)$ è un atomo

Definiamo ora le formule ben formate di un linguaggio logico predicativo, cioè l'insieme di espressione che possono essere costruite combinando termini e atomi attraverso connettivi e quantificatori.

12.3.3 Formule ben formate

L'insieme delle formule di L è l'insieme induttivo definito come segue:

- Ogni atomo è una formula
- Se A è una formula, $\neg A$ è una formula
- Se o è un connettivo binario, A e B due formule, AoB è una formula
- Se A è una formula, x una variabile, $\forall xA$ e $\exists xA$ sono formule

Le formule della logica predicativa (di primo ordine) si valutano su variabili individuali e singole e non su funzioni ed insiemi.

12.3.4 Chiarimenti sugli operatori

Gli operato della logica predicativa hanno la seguente precedenza:

$$\forall \exists \neg \wedge \vee \implies \iff$$

In caso di parità di precedenza si associa a destra (come nella logica proposizionale).

La scrittura $\forall xy$ è equivalente alla scrittura $\forall x \forall y$.

Il campo di azione del quantificatore $\forall x A$ è A, il campo di azione del quantificatore $\exists x A$ è A.

12.4 Variabili libere e legate

In una formula non atomica le variabili possono essere quantificate attraverso il quantificatore universale $\forall$ o quello esistenziale $\exists$.

Ad esempio in $\forall x Cane(x) \implies Animale(x)$ la variabile x è quantificata.

Ma le variabili potrebbero anche non essere quantificate ad esempio nella formula predicativa $Cibo(t) \land Buono(t)$. In questo caso la variabile è detta **libera** perché occorre senza essere quantificata.

Ovviamente la seconda frase non ha un vero e proprio significato perché deve essere contestualizzata (ad esempio dire che tipo di cibo sia 't').

Generalizzando, una variabile libera denota un individuo all'interno di un contesto.

Un atomo si dice chiuso se non contiene variabili.

12.4.1 Definizione insieme di variabili di un termine

L'insieme $var(t)$ delle variabili di un termine *t* è definito come segue:

- $var(t) = \{t\}$ se t è una variabile
- $var(t) = \emptyset$ se t è una costante
- $var(f(t_1, \ldots, t_n)) = \bigcup var(t_i)$ con i che varia tra 1 e n

Esempio L'insieme $var(t)$ delle variabili di una formula atomica $R(t_1, \ldots, t_n)$ è:

- $var(R(t_1, \ldots, t_n)) = \bigcup var(t_i)$ con i che varia tra 1 e n

12.4.2 Definizione di variabile che occorre libera

L'occorrenza libera di una variabile in una formula è definita sulla struttura della formula in modo induttivo:

- Se A è un atomo, x occorre libera in A se x occorre in A
- x occorre libera in $\neg A$ se x occorre libera in A
- x occorre libera in AoB se x occorre libera in A o x occorre libera in B
- x occorre libera in $\forall z A$ ed in $\exists z A$ se x occorre libera in A e $x \neq z$

12.4.3 Variabili legate

Una variabile è detta legata o vincolata se non occorre libera.

Esempio

$$\forall x(P(x) \implies Q(x, y))$$

Variabili legate: x, perchè nel campo di esistenza di $\forall x$

Variabili libere: y

13 | Semantica Logica Predicativa

Con la logica predicativa (che ricordiamo essere una logica del primo ordine) possiamo andare a rappresentare realtà del mondo reale che non erano rappresentabili attraverso la logica proposizionale.

Nella logica del primo ordine possiamo interpretare i vari simboli che compongono il linguaggio attraverso l'utilizzo di strutture matematiche: oggetti che permettono di trasformare formule in espressioni con significato specifico relativamente alla realtà che si sta rappresentando.

13.1 Funzione di interpretazione

Ad esempio la struttura per un linguaggio L ci deve fornire un qualche tipo di funzione che assegni al quantificatore $\forall$ (o a $\exists$) un insieme non vuoto di elementi su cui lavorare. Questa funzione è detta funzione di interpretazione e l'insieme non vuoto di elementi è detto dominio di interpretazione.

Una struttura per il linguaggio L è una coppia $U = \langle D, I \rangle$ dove:

- D è un insieme non vuoto chiamato Dominio di U
- I è una funzione di interpretazione che:
  - ad ogni simbolo di costante c associa un elemento $c^I \in D$
  - ad ogni simbolo di funzione n-aria f associa una funzione $f^I : D^n \to D$
  - ad ogni simbolo di predicato n-ario P associa una relazione n-aria $P^I \in D^n$

Ad ogni parametro di L viene quindi assegnato un preciso significato.