Strutture Relazionali, Grafi e Ordinamenti

course	Fondamenti dell'Informatica
completed?	☐
date	
tags	
links & files	06-RelRepresentation.pdf 07-RelEquiv-HO.pdf 08-Grafi-HO.pdf 09-Alberi-HO.pdf 10-Ordini-HO.pdf 11-Boole-HO.pdf
Property	Lezione 6

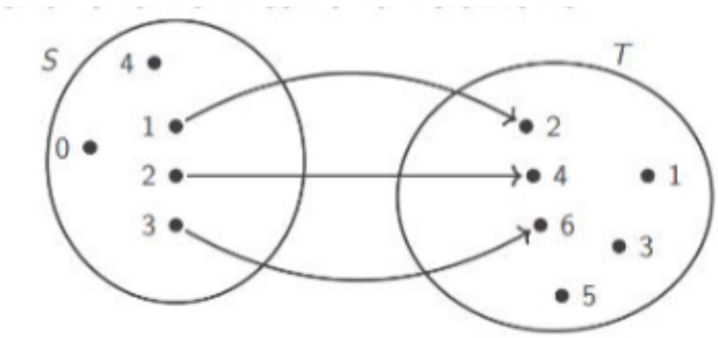
Relazioni

▼ Come si possono rappresentare le relazioni?

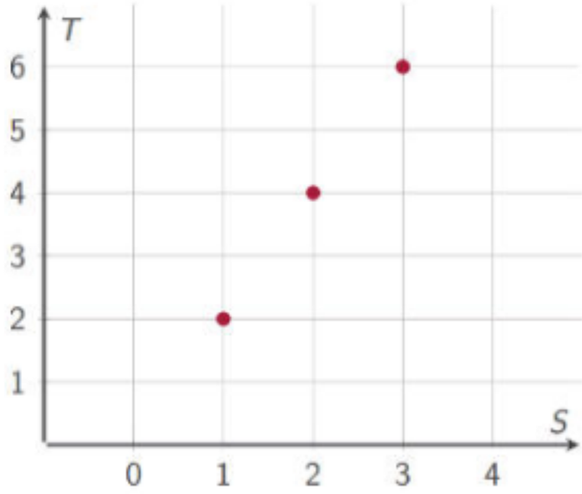
- Rappresentazione per elencazione

  R={(1,2), (2,4), (3,6)}

- Rappresentazione sagittale



- Rappresentazione tramite diagramma cartesiano



- Rappresentazione tramite tabella (Matrice booleana)

matrice di adiacenza

S \ T	1	2	3	4	5	6
0						
1		×				
2				×		
3						×
4						

Con 1 e 0

▼ Come si rappresentano le relazioni in un insieme?

Una relazione R contenuta in S x S è detta relazione in S.

La rappresentazione è un grafo, costituito da nodi collegati fra loro da frecce (o spigoli)

coppie	grafo
$\langle x, y \rangle \in R$	
$\langle x, x \rangle \in R$	
$\langle x, y \rangle \in R$ e $\langle y, x \rangle \in R$	

▼ Quali sono le proprietà delle relazioni?

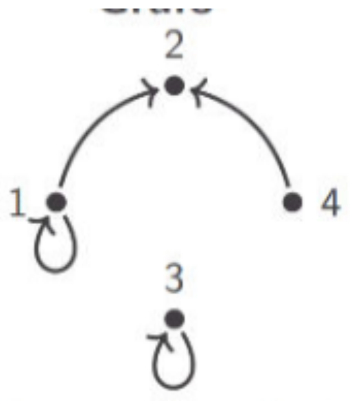
- **Riflessiva** se ci sono i cappi, la casella della diagonale è 1.
- **Irriflessiva** se non ci sono i cappi, la casella della diagonale è 0.
- **Simmetrica** se c'è avanti e indietro, la tabella è simmetrica rispetto alla diagonale.

	1	2	3	4
1	×	×		
2	×			×
3			×	
4		×		

- **Asimmetrica** se non c'è avanti e indietro, per ogni 1 la casella opposta è 0.
- **Antisimmetrica** per ogni freccia non c'è l'inverso, escluso i cappi.

Per ogni elemento, escludendo la diagonale

	1	2	3	4
1	×	×		
2				
3			×	
4		×		



- **Transitiva** se c'è uno con un altro e l'altro con un altro, uno è collegato con quell'altro là

▼ Quali sono le operazioni su matrici booleane?

- **join** M U N → la casella è 1 se la matrice M ha casella 1 ***OPPURE*** la matrice N ha la casella 1

$$\ell_{ij} = \begin{cases} 1 & m_{ij} = 1 \text{ o } n_{ij} = 1 \\ 0 & \text{altrimenti} \end{cases}$$

- **meet** M M N → la casella è 1 se la matrice M ha casella 1 ***E*** la matrice N ha la casella

$$\ell_{ij} = \begin{cases} 1 & m_{ij} = 1 \text{ e } n_{ij} = 1 \\ 0 & \text{altrimente} \end{cases}$$

- **prodotto booleano** → le due matrici devono avere un lato con la stessa lunghezza.
  ES: n x m e m x p. Il loro prodotto è L n x p.

$$\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \odot \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & \boxed{1} & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

La composizione di relazioni è il prodotto di matrici booleane.

▼ Cos'è una **relazione di equivalenza**? importante!

Relazioni che si comportano come la uguaglianza tra oggetti. Non esistono differenze tra due elementi in una relazione di equivalenza.

- **Definizione:** *Riflessiva, simmetrica, transitiva*

  Se $f : A \rightarrow B$ è una funzione totale, allora la relazione

  $$R := \{\langle x, y\rangle \in A \times A \mid f(x) = f(y)\}$$

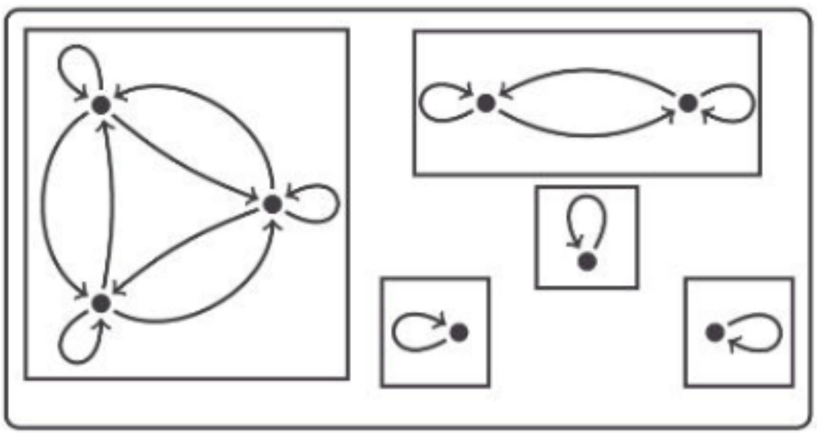
  è una **relazione di equivalenza.**

- **Rappresentazione sagittale:** *grafi totalmente collegati tra di loro e localmente totali.*

- **Classe di equivalenza**: Sia s un insieme e R una relazione di equivalenza su S.
  Ogni elemento x € S definisce una classe di equivalenza:

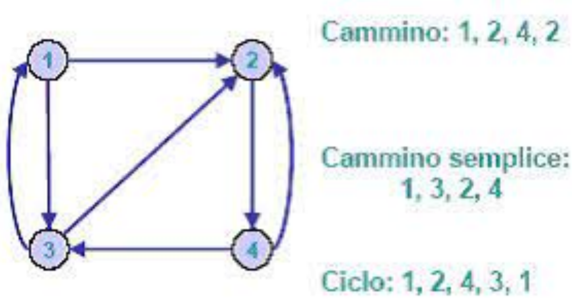
  $$[x]_R = \{y \in S \mid \langle x, y\rangle \in R\}$$

La famiglia di insiemi dove gli elementi sono le classi di equivalenza di A, è chiamato l’insieme quoziente di A rispetto a R. L’insieme quoziente è una partizione di A.



Ogni componente connesso è una classe di equivalenza.
L'insieme quoziente ha 5 insiemi: uno con tre elementi, uno con due elementi, e tre singoletti
R divide l'insieme in classi di equivalenza e da lì il nome "quoziente"

Grafi



▼ Cos'è un grafo?

Un grafo è definito per:

- Un insieme di nodi (o vertici)
- Collegamenti tra nodi che possono essere:
  - *Orientati (archi)* → *Grafo orientato*
  - *Non orientati (spigoli)* → *Grafo non orientato*
- (Facoltativo) Dati associati ai nodi e collegamenti (*etichette*)
  - → *Grafo etichettato*

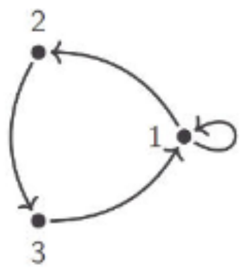
Notazione: G=(V,E)
G è il grafo, V insieme dei nodi, E è la relazione.

▼ Come si rappresenta?

Un grafo viene rappresentato disegnando punti per i nodi e segmento o curve per i collegamenti tra i nodi.

Importanti (per confronto con gli altri metodi di rappresentazione)

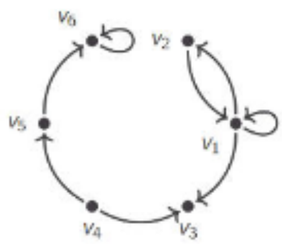
- **Posizione o forma dei nodi e dei collegamenti sono irrilevanti**
- Soltanto la loro esistenza definisce il grafo.



▼ Cosa rappresenta?

I grafi possono rappresentare **relazioni binarie.**

v_1	v_1
v_1	v_2
v_1	v_3
v_2	v_1
v_4	v_3
v_4	v_5
v_5	v_6
v_6	v_6



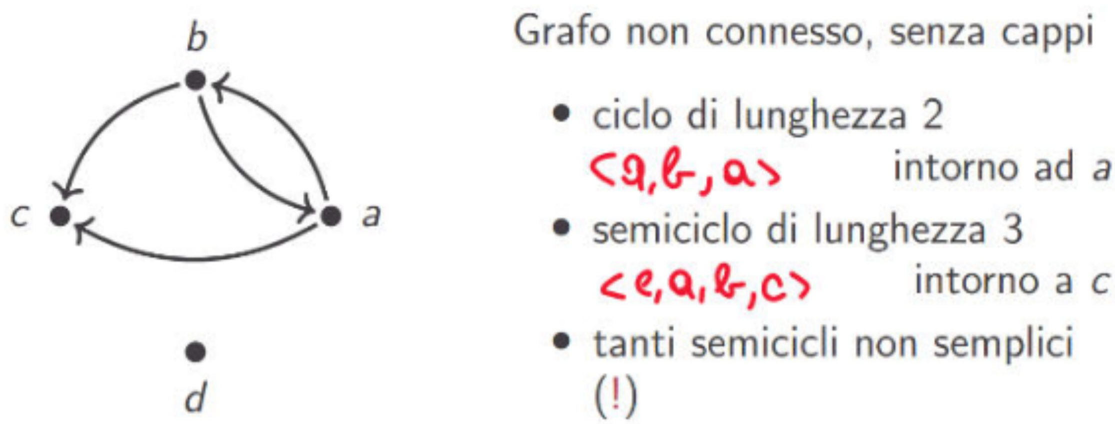
$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

▼ Quali sono le sue caratteristiche? → importante

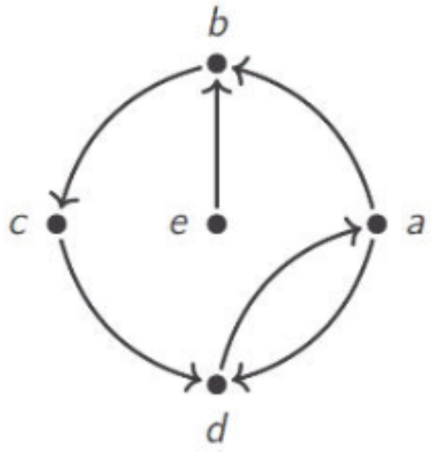
- ***Nodo***
  - *Sorgente*: se non ha archi entrati (il suo grado di ingresso è 0)
  - *Pozzo*: se non ha archi uscenti (grado di uscita 0)
  - *Isolato*: se non ha archi né uscenti né entranti (è sia sorgente che pozzo).
  - *Adiacenti*: i nodi v e w sono adiacenti se c'è un arco tra v e w, **in qualunque direzione. Questo arco è incidente su v e w.**
- ***Grado***
  - *Grado di uscita*: numero di archi uscenti dal nodo
  - *Grado di ingresso:* numero di archi entranti nel nodo
  - *Adiacenti*: i nodi v e w sono adiacenti se c'è un arco tra v e w, **in qualunque direzione. Questo arco è incidente su v e w. Il grado di v è il numero di nodi adiacenti a v.**
- ***Cammino:*** sequenza finita di nodi (tupla ordinata di nodi). Tra ogni nodo che prendiamo in considerazione, esiste un arco uscente da Vi ed entrante in

Vi+1. Se ci sono frecce, devo seguire le frecce altrimenti non è un cammino. Ricorda che è ordinato.
In un cammino non si possono ripetere nodi ed archi.

- ***Semicammino***: sequenza finita di nodi (tupla ordinata di nodi) ma dimentichiamo della direzione.
  - *Lunghezza di un (semi)cammino:* il numero di archi che lo compongono (numero nodi - 1)
  - *(Semi)cammino semplice:* tutti i nodi nella sequenza sono diversi e non si ripetono. In un semi cammino si possono ripetere nodi ed archi.
  - *Grafo connesso:* esiste sempre un semicammino tra due nodi qualsiasi.
- ***Ciclo:*** un ciclo intro al nodo v è un cammino tra v e v.
- ***Semiciclo:*** Un semiciclo intorno al nodo v è un semicammino tra v e v.
- ***Cappio:*** Un cappio intorno a v è un ciclo di lunghezza 1.



- ***Distanza:*** La distanza tra v e w è la lunghezza del cammino più corto tra v e w.
  - La distanza tra v e v è 0;
  - Se non esiste nessun cammino tra v a w allora la distanza è infinita.
  - In un grafo ordinato, la distanza da v a w non è sempre uguale alla distanza da w a v.

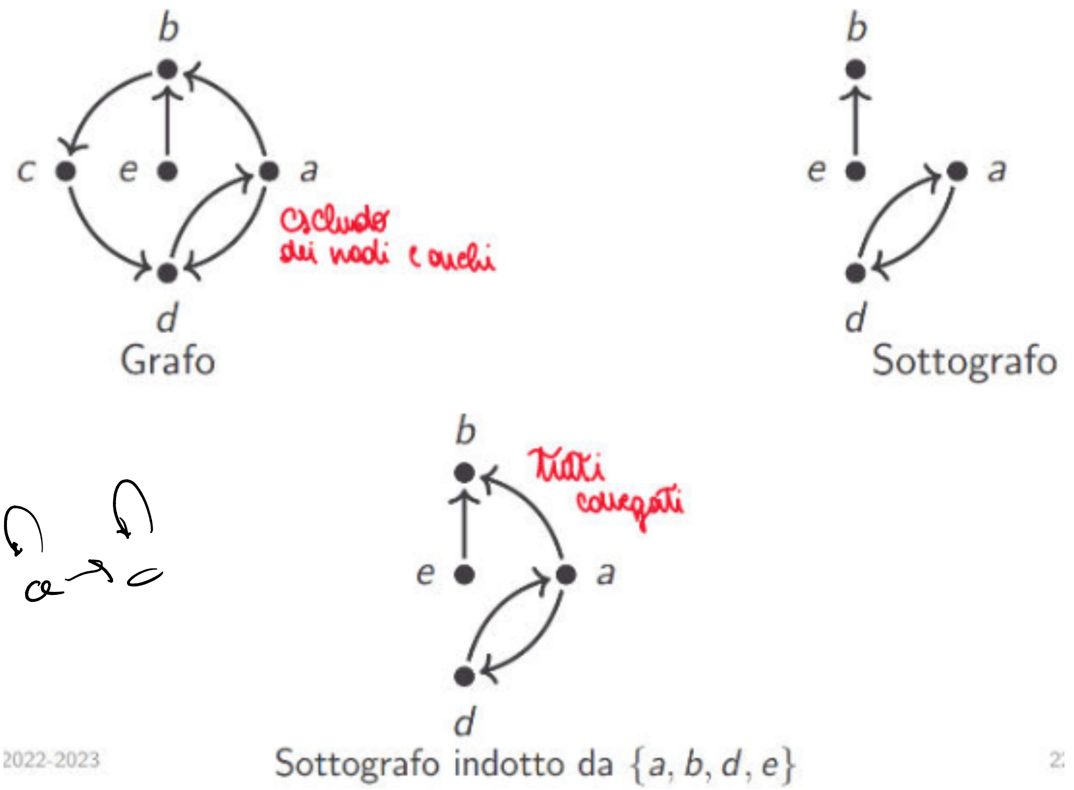


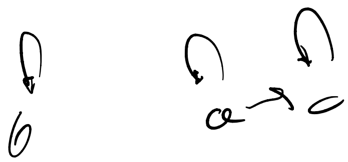
- distanza da a a b: 1
- distanza da a a d: 1
- distanza da b ad a: 3
- distanza da d ad a: 1
- distanza da e a d: 3
- distanza da a ad e: ∞

- ***Connettività:***
    - Grafo *connesso:* G= (V, E) è connesso se per ogni v, w ∈ V esiste un **semicammino** da v a w
    - *Grafo fortemente connesso:* G=(V,E) è fortemente connesso se per ogni v, w ∈ V esiste un **cammino** tra v a w.
        - Esiste sempre un ciclo che visita ogni nodo (non necessariamente semplice).
        - Non ci sono né sorgenti né pozzi.

▼ Quali sono i tipi di grafo?

- ***Grafo orientato:*** Ci sono relazioni binarie
- ***Grafo non orientato:*** Ci sono relazioni simmetriche e gli archi sono rappresentati come coppie non ordinate (v,w) [(v,w)=(w,v)]. Togliamo le frecce agli archi.
- ***Sottografo:*** Si ottiene togliendo nodi e/o archi dal grafo.
    - *Sottografo indotto:* è sottografo dove consideri gli archi solo di certi nodi indotti.

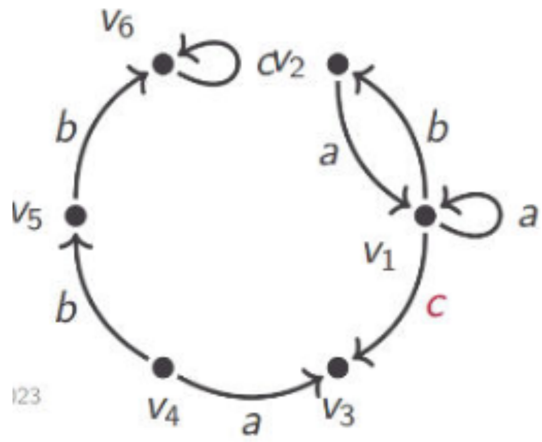




importante →

- ***<u>Grafo aciclico orientato (DAG):</u>*** Grafo orientato senza cicli. Non ci sono frecce che tornano indietro aka non esiste cammino da un nodo a sé stesso (Esempio: Albero)

- ***<u>Grafi etichettati:</u>*** è una tripla G=(V,E,l) cone l: E → L è una funzione totale che associa ad ogni arco e ∈ E una etichetta da un insieme L. Aka diamo una etichetta ad ogni arco del grafo.



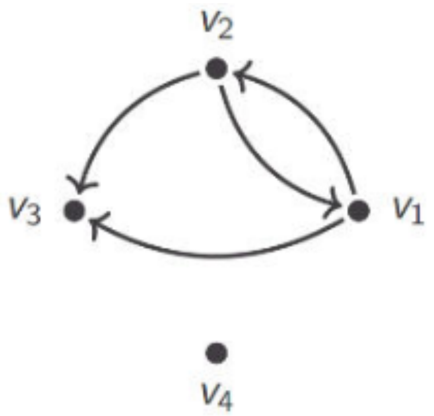
Un grafo etichettato può rappresentare una relazione ternaria (e viceversa).

importante!

- ***Grafo completo:*** un grafo completo è **irriflessivo** e collega ogni nodo con tutti gli altri nodi, ma non con sè stesso.

▼ Cos'è la matrice di adiacenza?

La matrice di adiacenza di un grafo è la matrice booleana della relazione E.



$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

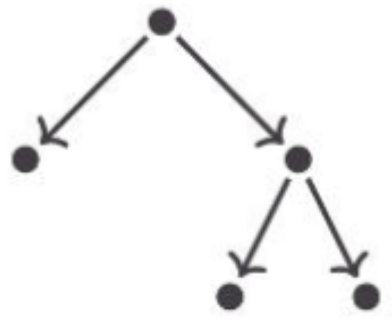
La matrice di adiacenza di grafi non orientati è sempre simmetrica.

▼ Cosa sono i grafi isomorfi?

Due grafi sono isomorfi se mantengono la stessa struttura del grafo ma sostituisci i nomi dei vertici per quelli del secondo grafo.

Essenzialmente, due grafi isomorfi sono in realtà lo stesso grafo con i nodi rinominati

Albero



- Cos'è un albero?

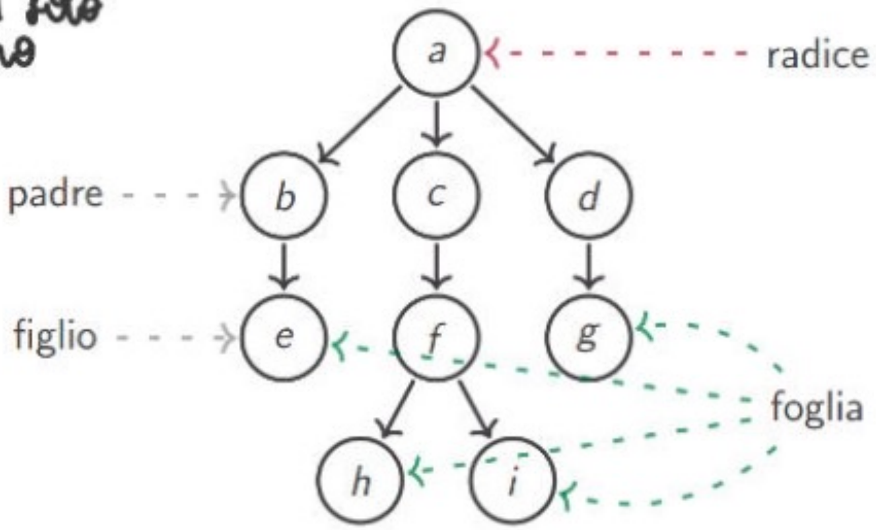
L'albero è una DAG, un grafo orientato senza cicli. Non esiste nessun cammino da un nodo a sé stesso.

- *Radice dell'albero*: nodo sorgente;
- Ogni nodo diverso dalla radice *ha un solo arco entrante*.

- Quali sono gli elementi di un albero?
  - Nodo sorgente: nodo che non ha archi entranti;
  - Foglia o nodo esterno: nodo pozzo di un albero;
  - Nodi interni: tutti gli altri nodi.

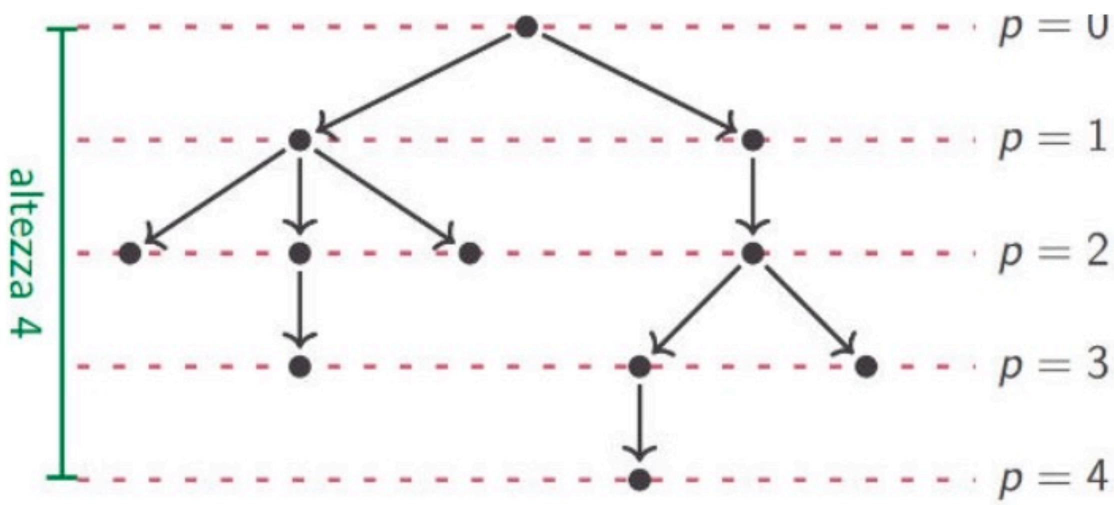
- Quali sono le proprietà?
  - Il **grado di ingresso** di un nodo è 1 se non è la radice, 0 se radice. Il grado di uscita di un nodo non ha restrizioni.
  - Esiste **un solo cammino** dalla radice a un nodo.
  - Un albero non può essere **mai vuoto**.
  - Se un albero è finito, allora **esiste almeno una foglia** che può essere la radice.
  - I nodi intermedi sono **sia padre che figlio**.
  - In un albero **esiste esattamente un cammino** da radice a qualsiasi nodo.
  - **Ascendente**: nodo che si trova prima; **Discendente**: nodo che si trova dopo.
  - **Padre**: nodo che si trova subito prima; **Figlio**: nodo che si trova subito dopo.

c'è sempre uno e un solo cammino



▼ Quali sono le sue caratteristiche?

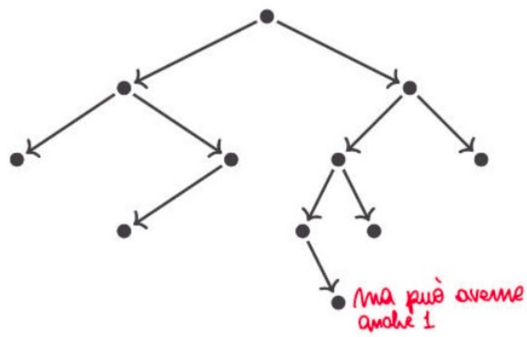
- **Profondità**: lunghezza del cammino dalla radice a v.
- **Altezza**: profondità massima dei suoi nodi.



▼ Quali sono i tipi di albero?

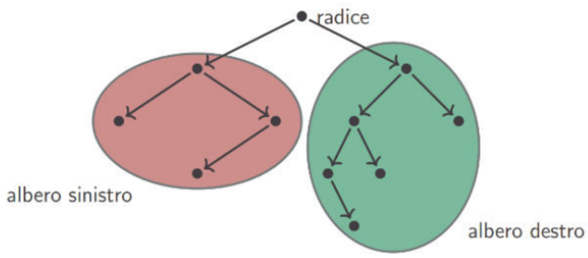
▼ **Albero binario**

Albero dove ogni nodo ha al massimo due figli. I figli di un nodo sono ordinati (figlio sinistro e figlio destro).



Un albero binario è una ***struttura ricorsiva*** composta da:

- Un nodo (radice);
- Un albero binario sinistro (eventualmente vuoto);
- Un albero binario destro (eventualmente vuoto).



Proprietà

- Ha al massimo 2^p nodi di profondità p.
- Ad altezza n, ha al massimo

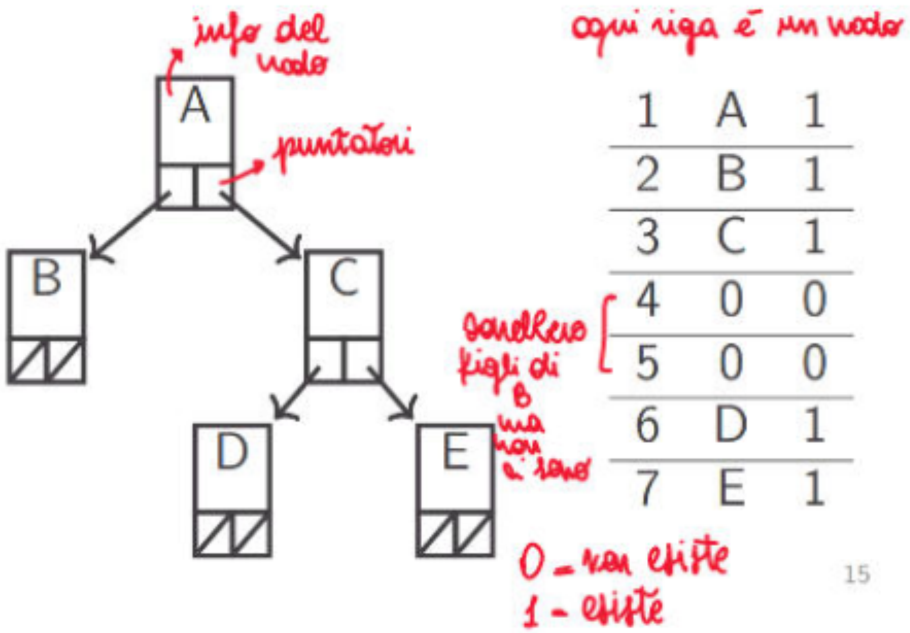
$$\sum_{i=0}^{n} 2^i = 2^{n+1} - 1 \text{ nodi}$$

Rappresentazione

Possiamo rappresentare un albero binario sia come:

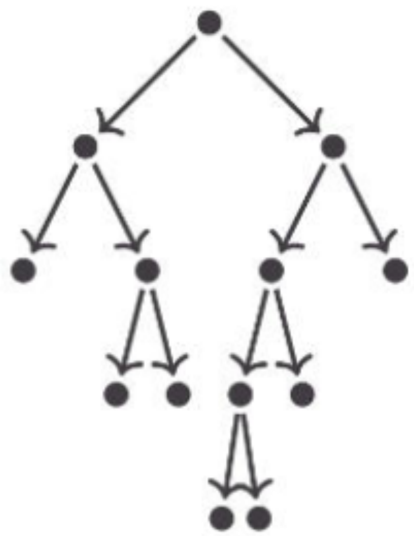
- Un grafo :)
- Una collezione di nodi, dove la radice è segnata e ogni nodo ha due puntatori (Alle radici degli alberi sinistro e destro)

- Una tabella con 2^(n+1) + 1 righe, dove n è la altezza dell'albero. Si legge prima l'albero sinistro fino alle foglie poi si sposta più a destra.



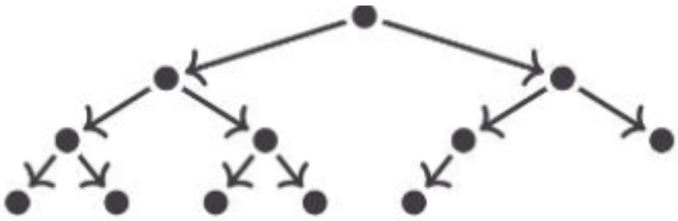
Albero binario pieno

Un albero binario è pieno se ogni nodo interno ha due figli.



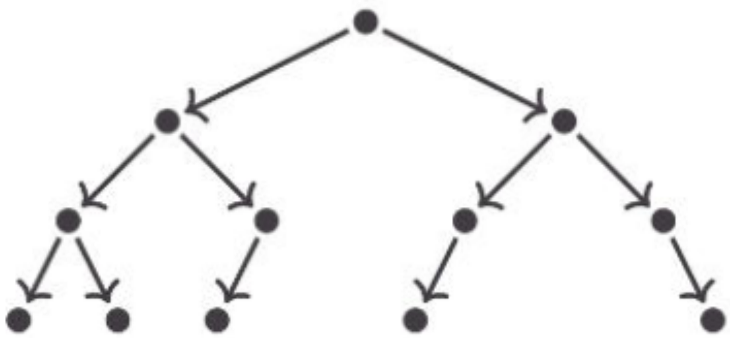
Albero binario completo

- Ad ogni profondità i, compreso tra 0 e n (altezza dell'albero) ci sono 2^i nodi;
- L'ultimo livello è riempito da sinistra a destra. I nodi vuoti sono soltanto sulle ultime righe.



▼ **Albero bilanciato**

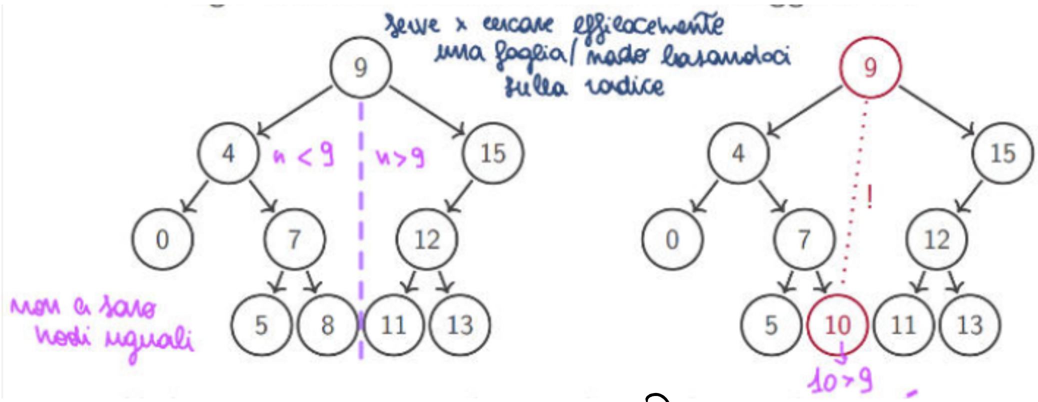
Un albero binario è bilanciato se per ogni nodo v: ***la differenza tra il numero di nodi dell'albero sinistro di v e l'albero destro di v è al massimo 1.***



▼ **Albero di ricerca**

Albero binario G=(V,E) tale che per ogni nodo z:

- ogni nodo dell'albero sinistro di z è minore a z
- ogni nodo dell'albero sinistro di z è maggiore a z.



$A = \{a, b, c, d\}$ $\quad R \subseteq A \times A$ $\quad R = \{\langle a,a\rangle, \langle a,b\rangle$

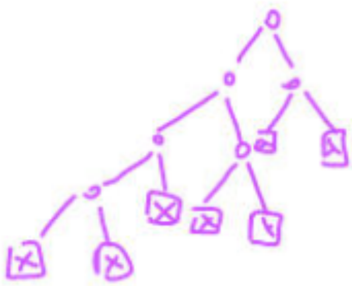
$R^{-1} = \{\langle a,b\rangle, \langle b,a\rangle$

$R^{-1} \circ R = \langle a,b\rangle \langle b,a$

▼ **Albero finito**

Un albero è infinito perché ci possono essere foglie vuote. Nel nostro caso, negli esercizi ci sono sempre alberi finiti altrimenti non si potrebbero svolgere gli esercizi.

Un albero finito da sempre almeno una foglia. Per massimizzare il numero di foglie dobbiamo avere un albero pieno. Un albero pieno con n nodi interni ha n+1 foglie. Basta sostituire i puntatori vuoti per foglie speciali per formare l'albero pieno se l'albero non è pieno.



▼ Cos'è l'attraversamento?

Un attraversamento è un processo che visita tutti i nodi di un albero. Solitamente di un ordine particolare.

Tipi di attraversamento

▼ ***In profondità***

Esplora ogni ramo dell'albero fino in fondo (figli prima del fratelli).

▼ ***In ampiezza***

Esplora prima i nodi più vicini alla radice (fratelli prima dei figli).

▼ Cos'è l'enumerazione?

Un attraversamento che elenca ogni nodo esattamente una volta è una enumerazione.

Ci sono **tre tipi diversi di ordini** in profondità basati su quando enumeriamo un elemento:

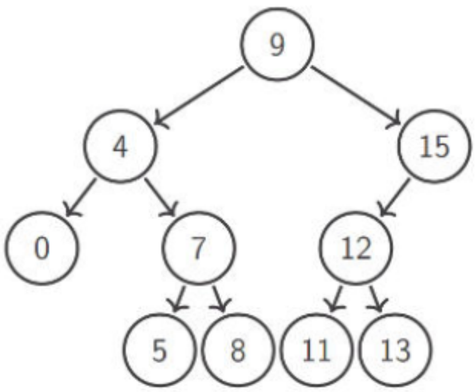
- L per sinistra;
- R per destra;
- V per enumerazione.

Tipi di enumerazione

▼ *Enumerazione in profondità*

Ci sono **tre tipi di enumerazione in profondità**:

1. In **preordine** (VLR) : si visita un nodo prima di visitare i figli;
2. In **ordine** (LVR) : si visita l'albero sinistro, poi il nodo, poi l'albero destro;
3. In **postordine** (LRV) : si visitano prima i figli e poi il nodo.



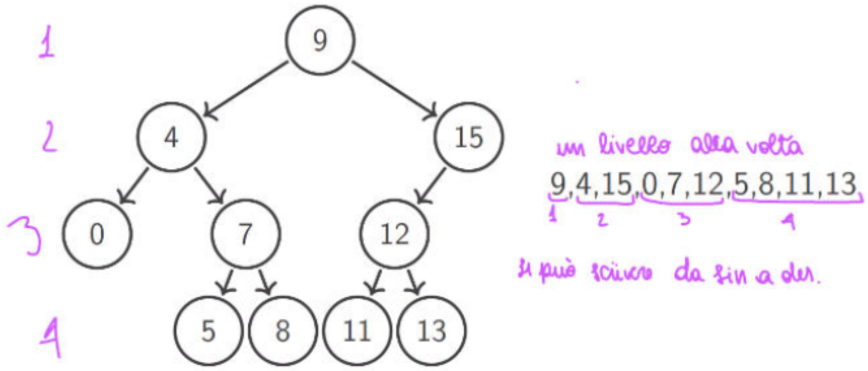
VLR ~> preordine
9,4,0,7,5,8,15,12,11,13

LVR ~> si cerca il più sinistra, albero sinistro
0,4,5,7,8,9,11,12,13,15

LRV
0,5,8,7,4,11,13,12,15,9

▼ *Enumerazione in ampiezza*

Visita tutti i nodi ad una profondità prima di esplorare altri livello dell'albero (**un livello alla volta e si scrive da sinistra a destra**).



Un livello alla volta e si scrive da sinistra a destra

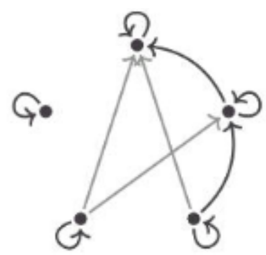
Ordinamenti

▼ Cos'è un ordinamento?

Gli elementi di un insieme hanno una struttura d'ordine. Un ordinamento è un tipo particolare di relazione fra elementi caratterizzati da certe proprietà.

▼ Quali sono i tipi di ordinamento? importante!

- **Preordine:** R è riflessiva e transitiva;
- **Ordine parziale**: R è riflessiva, antisimmetrica e transitiva. Un insieme parzialmente ordinato viene chiamato poset.



ordine parziale

rifless., transitiva

- **Ordine stretto:** R è irriflessiva, transitiva, quindi anche asimmetrica.
- **Ordine totale:** ordine parziale fortemente connesso.

per ogni $x, y \in S$: $\quad x \leq y$ oppure $y \leq x$

ordine totale
ogni 2 elementi hanno una relazione orient. + transitivo

- **Ordine totale stretto:** ordine stretto connesso.

$$\text{per ogni } x, y \in S \text{ tali che } x \neq y: \qquad x < y \ \text{ oppure } \ y < x$$

  In questo ordine, per ogni x,y appartenente ad S, si soddisfa una fra queste proprietà tricotomiche.

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

ordine totale stretto
tolgo tutti cappi

Differenza tra ordine totale e ordine totale stretto: ordine stretto è riflessivo mentre ordine totale è irriflessivo.

▼ Cosa sono i prodotti di ordinamenti?

Poset prodotto: abbiamo due poset. Il prodotto tra i due poset è anch'esso un poset.

Ordine lessicografico: paragona tuple di elementi posizione per posizione. L'ordine utilizzato per mettere in ordine alfabetico il lessico.

- *Ordine lessicografico*: ⟨0,1⟩≤⟨1,0⟩

$$\langle 1,3\rangle \le_{lex} \langle 1,4\rangle$$

$$\langle a,z\rangle < \langle b,a\rangle$$

non importa il resto.

- *Poset prodotto*: ⟨0,1⟩≰⟨1,0⟩

▼ Quali sono gli elementi di un ordinamento? importante!

- **Copertura**: elemento minimo più grande di x.
- **Elemento estremale**: un poset può avere nessuno, uno o tanti elementi minimali e massimali.
    - ***Minimale***: se non esiste un elemento diverso da s tale che s' ≤ s.
    - ***Massimale***: se non esiste un elemento diverso da s tale che s' ≥ s.

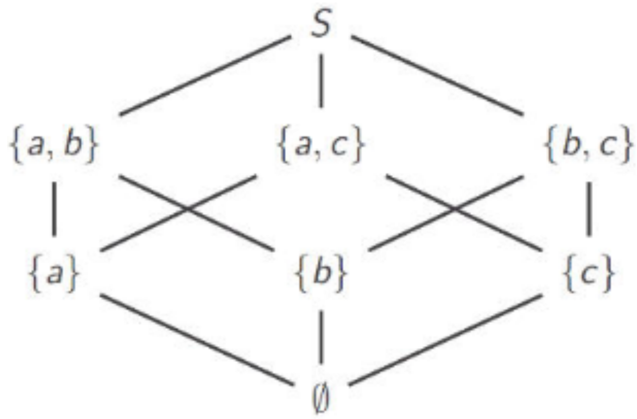
Dato un poset (S,≤) e un insieme X⊆S, un elemento s ∈ S è

- **Minorante di X** sse s ≤ x per ogni x ∈ X (ci possono essere tanti minoranti).
- **Massimo minorante di X** (⊓X) sse s' ≤ s per ogni minorante s' di X (il più grande dei minoranti di X).
- **Maggiorante di X** sse x ≤ s per ogni x ∈ X (ci possono essere tanti maggioranti).
- **Minimo maggiorante di X** (⊔X) sse s ≤ s' per ogni maggiorante s' di X (il più piccolo dei maggioranti).
- **Minimo di X** sse s = ⊓X ∈ X.
- **Massimo di X** sse s = ⊔X ∈ X.

▼ Quali sono le proprietà di un ordinamento?

- Ogni X ⊆ S ha al più un massimo minorante e un minimo maggiorante;
- Se ogni X ⊆ S ha un minimo, allora (S,≤) è un insieme ben ordinato (o ben fondato);
- Se esiste, ⊓S è il minimo di S, denotato da 0.
- Se esiste, ⊔S è il massimo di S, denotato da 1.

▼ Qual è il modo di rappresentare un ordinamento?



Un diagramma di Hasse è una rappresentazione compatta di un poset. Importante la posizione per rappresentare l'ordine e considera la riflessività e transitività implicite.

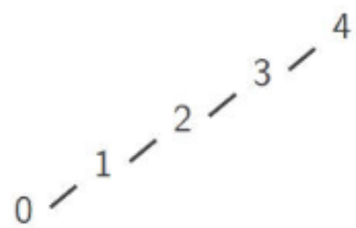
Un diagramma di Hasse è un grafo non orientato tale che per ogni x,y:

- Se $x \leq y$ allora x appare sotto di y;
- x e y sono collegati sse y è una copertura di x.

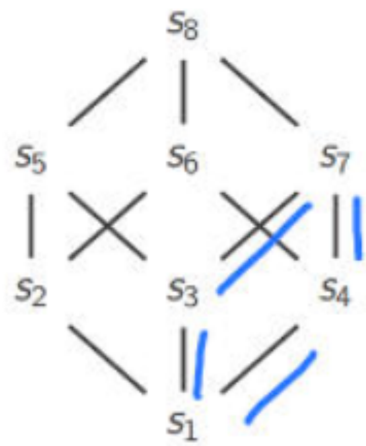
come realizzare un diagramma di Hasse.

Per realizzarne uno, si tolgono i cappi, chiusure transitive e le frecce. Il diagramma di Hasse di un ordinamento totale formerà sempre una catena.

Per esempio, $(\{0, 1, 2, 3, 4\}, \leq)$:



Reticolo



riflessivo, antisimmetrico, transitivo

▼ Cos'è un reticolo?

Un reticolo è un poset (S,≤) tale che per ogni x,y ∈ S:

- Esiste un minimo maggiorante x ⊔ y (join);
- Esiste un massimo minorante x ⊓ y (meet);

▼ Quali sono le proprietà?

- **Idempotenza**: a ⊔ a = a = a ⊓ a
- **Commutatività**: a ⊔ b = b ⊔ a
- **Associatività**:

  a ⊔ (b ⊔ c) = (a ⊔ b) ⊔ c

  a ⊓ (b ⊓ c) = (a ⊓ b) ⊓ c
- **Assorbimento**: a ⊔ (a ⊓ b) = a = a ⊓ (a ⊔ b)

Se (L, ≤) è un reticolo, allora per ogni a, b, c ∈ L:

- a ≤ a ⊔ b
- se a ≤ c e b ≤ c allora a ⊔ b ≤ c
- a ⊓ b ≤ a
- se c ≤ a e c ≤ b allora c ≤ a ⊓ b
- a ⊔ b = b sse a ≤ b
- a ⊓ b = a sse a ≤ b

Monotonicità di join e meet

Se a ≤ c e b ≤ d allora

- a ⊔ b ≤ c ⊔ d
- a ⊓ b ≤ c ⊓ d

▼ Quali sono i tipi di reticoli?

Completo sse per ogni M ⊆ L, ⊔M e ⊓M esistono. Se è completo, è limitato.

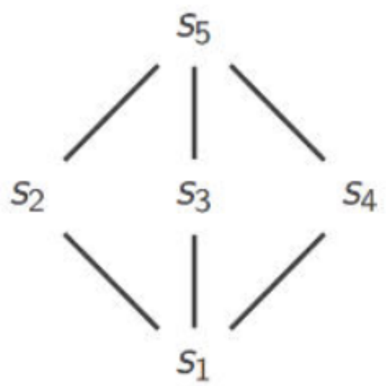
Limitato sse 1 = ⊔L e 0 = ⊓L esistono.

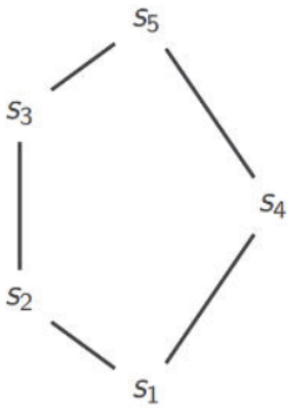
Ogni reticolo finito è completo e limitato. Ogni reticolo che trattiamo in esame è finito, altrimenti non potremmo fare questi esercizi. Concentrati su reticolo distributivo o non distributivo.

Distributivo sse meet e join distribuiscono fra di loro:

- a ⊓ (b ⊔ c) = (a ⊓ b) ⊔ (a ⊓ c)
- a ⊔ (b ⊓ c) = (a ⊔ b) ⊓ (a ⊔ c)

Non distributivo: i due reticoli non distributivi prototipici sono questi. I reticoli non distributivi hanno questa struttura all'interno.





▼ Cos'è il complemento di un reticolo?

Un elemento b ∈ L è il complemento di a sse

$a \sqcap b = 0$ e $a \sqcup b = 1$

Questo complemento è unico e se ogni elemento del reticolo ha un complemento, è un reticolo complementato. → lo si cerca in quelli con cui non hai un colleg. diretto

Per trovare il complemento devo trovare un elemento tale che il max tra E e l'altro elemento diano il max assoluto.

Algebra di Boole

▼ Da cosa nasce l'Algebra di Boole?

Boole prova a formalizzare le regole di ragionamento combinando proposizioni in base al loro valore di verità. Corrisponde alla prima formalizzazione delle operazioni logiche:

- Congiunzione ("e");
- Disgiunzione ("oppure");
- Negazione ("no").

Due valori di verità:

- vero: 1
- falso: 0

Presto generalizzò a strutture più complesse chiamate algebre di Boole.

▼ Cos'è un reticolo booleano?

cos'è un reticolo booleano

Un reticolo booleano è un reticolo

- limitato;
- distributivo;
- complementato (ogni oggetto ha complemento).

Un reticolo booleano definisce l'algebra di Boole con operazioni per disgiunzione, congiunzione e negazione. Invece di chiamare gli elementi, saranno 0 e 1.

▼ Quali sono le operazioni logiche?

- ***Disgiunzione*** è data dal join (una cosa è vera se almeno uno è vero) (⋁)
  Il massimo fra valori logici

$\vee$	0	1
0	0	1
1	1	1
0	0	0

- ***Congiunzione*** è data dal meet (una cosa è vera se entrambi gli elementi sono veri) (⋀)
  Il minimo fra valori logici

$\wedge$	0	1
0	0	0
1	0	1

- ***Negazione*** è data dal complemento (¬)
  L'opposto del valore logico

$\neg$	
0	1
1	0

▼ Quali sono le proprietà delle operazioni logiche?

- ⋀ e ⋁ sono idempotenti, commutative e associative;

- ¬ è involutivo; ¬¬x = x
- ⋀ e ⋁ distribuiscono fra di loro

  $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$

  $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$
- Soddisfano le leggi di De Morgan

  $\neg(x \wedge y) = \neg x \vee \neg y$

  $\neg(x \vee y) = \neg x \wedge \neg y$

▼ Cosa sono i circuiti logici?

Le operazioni booleane si possono implementare fisicamente su porte logiche.
Ci permettono di manipolare valori logici in applicazioni molto complesse.