

FONDAMENTI DELL'INFORMATICA

STRUTTURE RELAZIONALI, GRAFI E ORDINAMENTI (parte 4)

Stefania Bandini

FONDAMENTI DELL'INFORMATICA

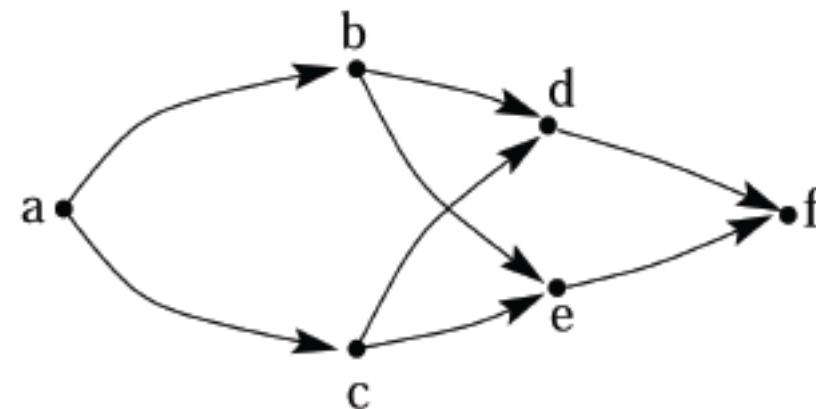
ALBERI

GRAFO DIRETTO ACICLICO (DAG)

Un *grafo diretto aciclico* (detto anche DAG, dall'inglese “Directed Acyclic Graph”) è un grafo diretto senza cicli.

Sia $R \subseteq S \times S$ dove $S = \{a, b, c, d, e, f\}$;

$$R = \{\langle a, b \rangle, \langle a, c \rangle, \langle b, d \rangle, \langle c, d \rangle, \langle b, e \rangle, \langle c, e \rangle, \langle d, f \rangle, \langle e, f \rangle, \}.$$



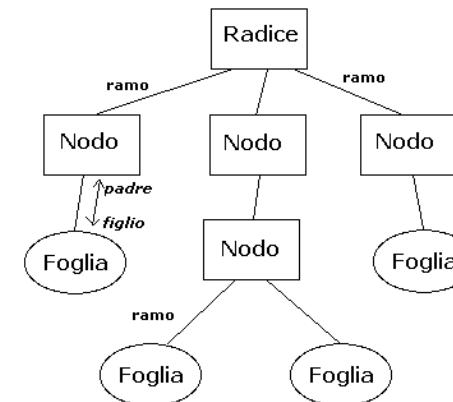
ALBERI

Un *albero* è un DAG connesso con un solo nodo sorgente (detto *radice* dell'albero) in cui ogni nodo diverso dalla radice ha un solo arco entrante.

I nodi privi di archi uscenti sono detti *foglie* dell'albero.

Per analogia con gli alberi genealogici i nodi intermedi si chiamano *padre*, *figli*, *fratelli*, *discendenti*, *avi*, con l'ovvio significato.

Di solito gli alberi vengono disegnati ponendo la radice in alto e le foglie in basso, in analogia con gli alberi genealogici. Quindi non è necessario disegnare le “punte” alle frecce.



FONDAMENTI DELL'INFORMATICA

ALBERI

Un **albero** o **struttura ad albero** (*tree* in inglese) è la struttura dati che si riconduce al concetto di albero con radice presente nella teoria dei grafi.

Un albero si compone di due tipi di sottostrutture fondamentali: il **nodo**, che in genere contiene informazioni, e l'**arco** che stabilisce un collegamento gerarchico fra due nodi: si parla allora di un **nodo padre** dal quale esce un arco orientato che lo collega ad un **nodo figlio**.

Ogni nodo può avere al massimo un unico arco entrante, mentre dai diversi nodi possono uscire diversi numeri di archi uscenti.

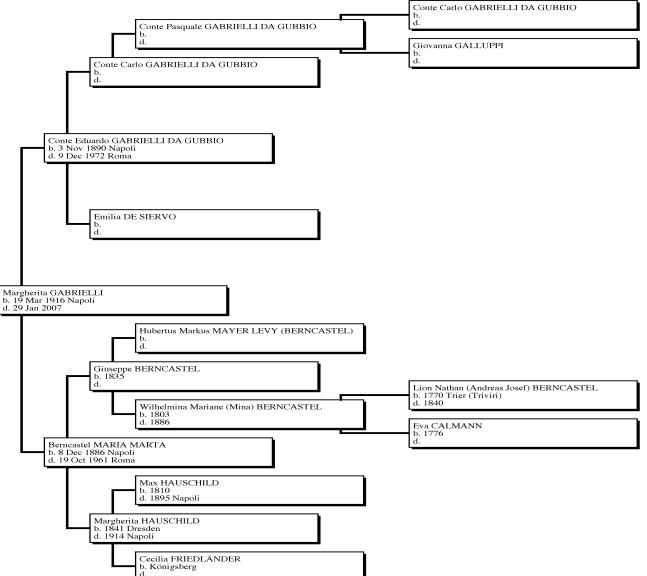
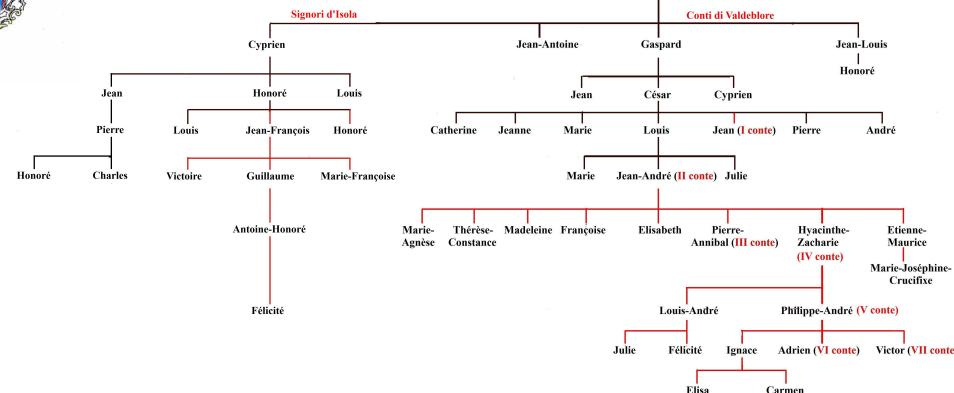
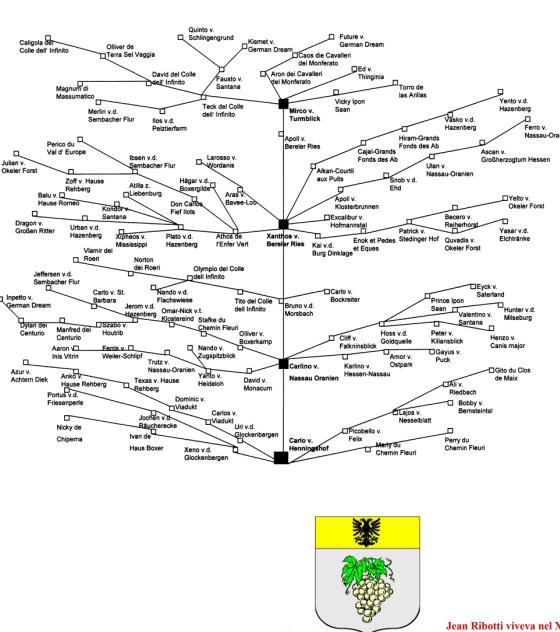
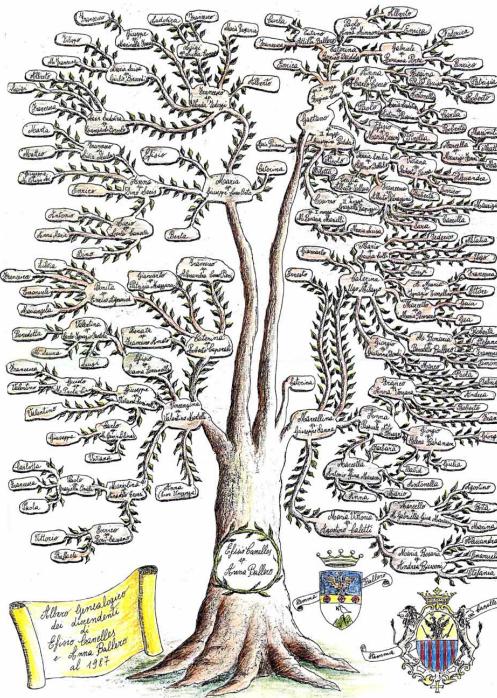
Un albero possiede un unico nodo privo di arco entrante: questo nodo viene detto **radice** (*root*) dell'albero.

Ogni nodo che non presenta archi uscenti, è detto **foglia** (*leaf node*); e in ogni albero finito, cioè con un numero finito di nodi, si trova almeno un nodo foglia.

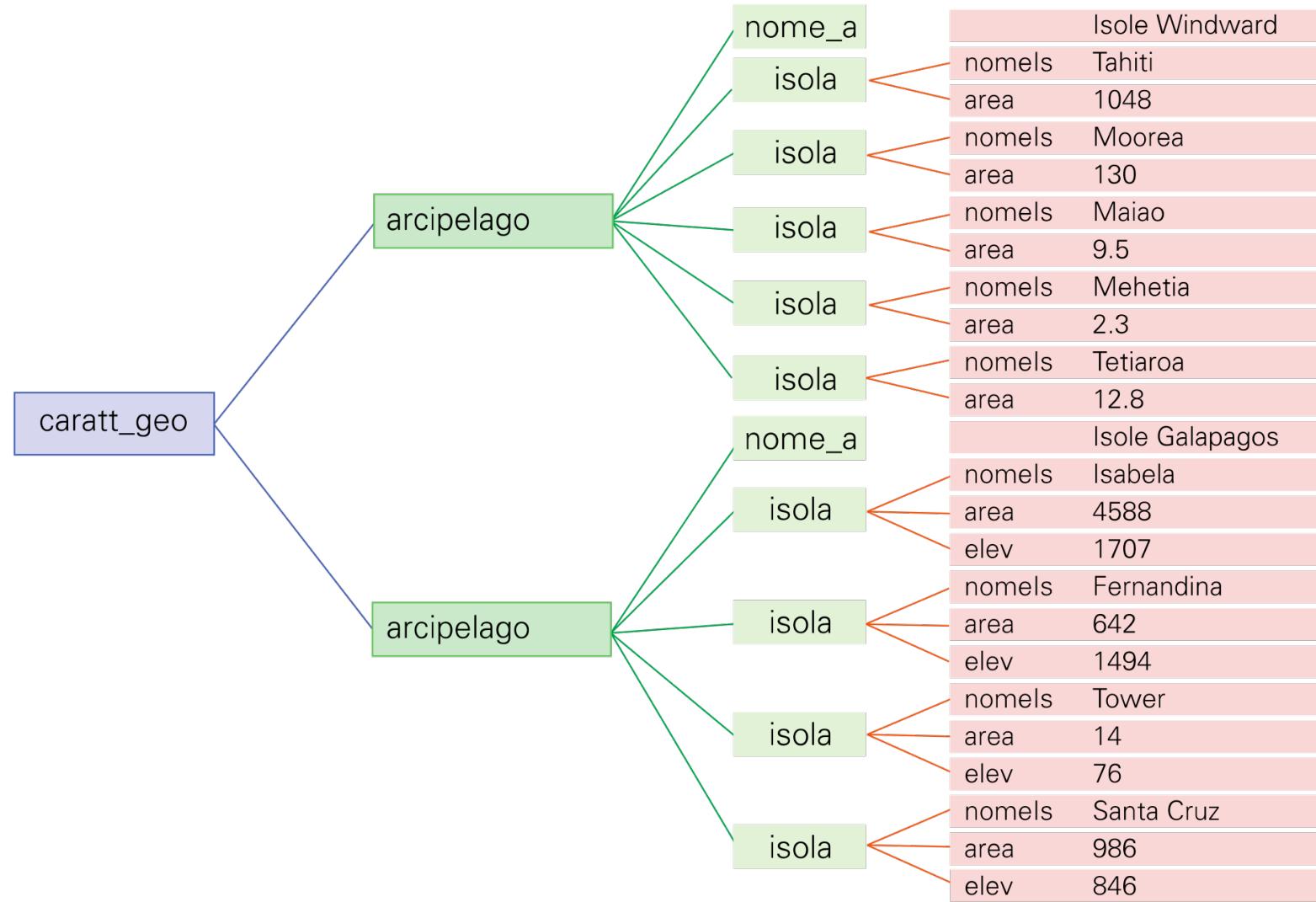
Un nodo può essere contemporaneamente padre (se ha archi uscenti) e figlio (se ha un arco entrante, ovvero se è diverso dalla radice).

FONDAMENTI DELL'INFORMATICA

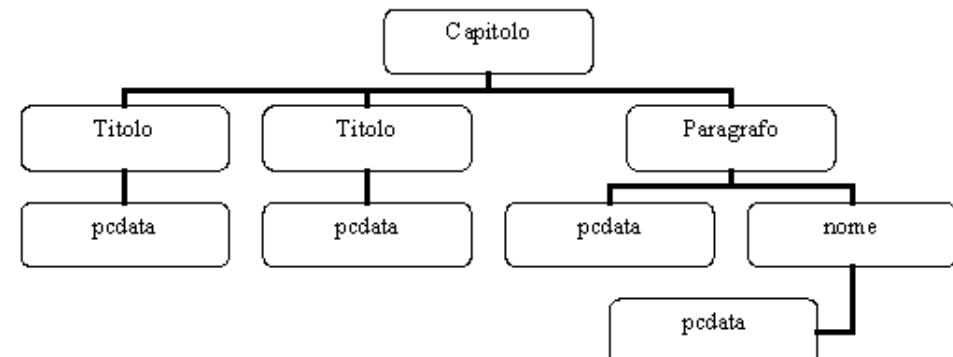
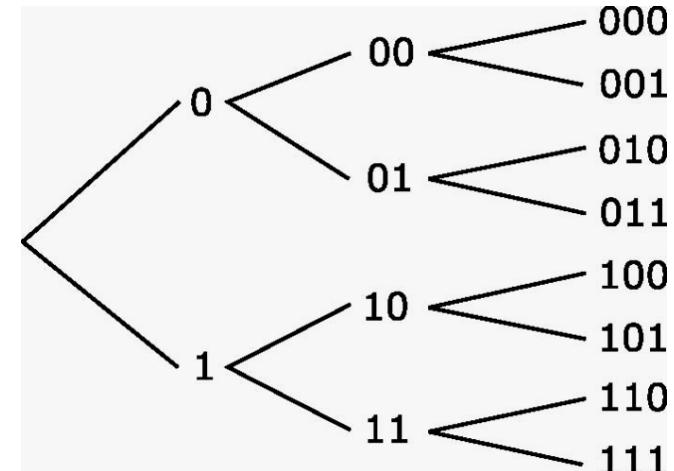
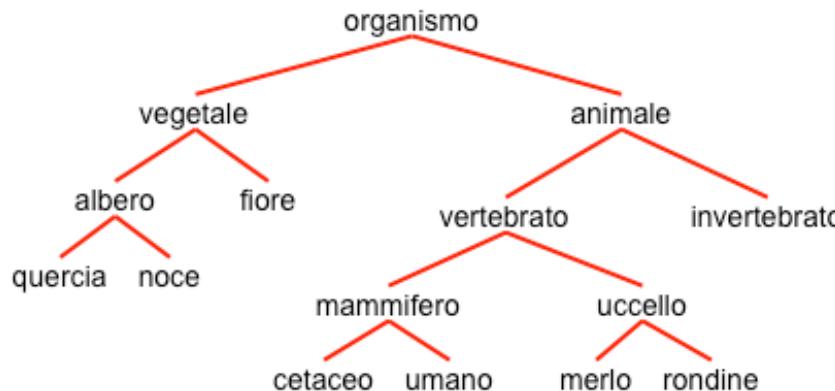
ALBERI - ESEMPI



ALBERI - ESEMPI



ALBERI - ESEMPI

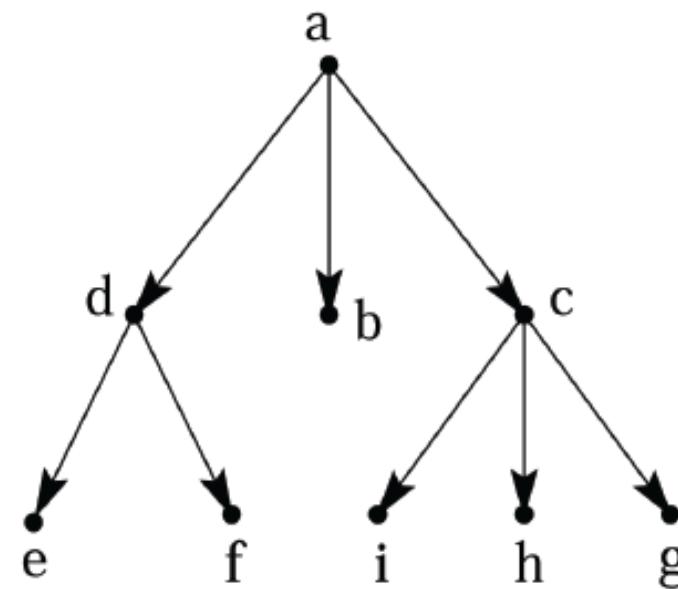


ALBERO

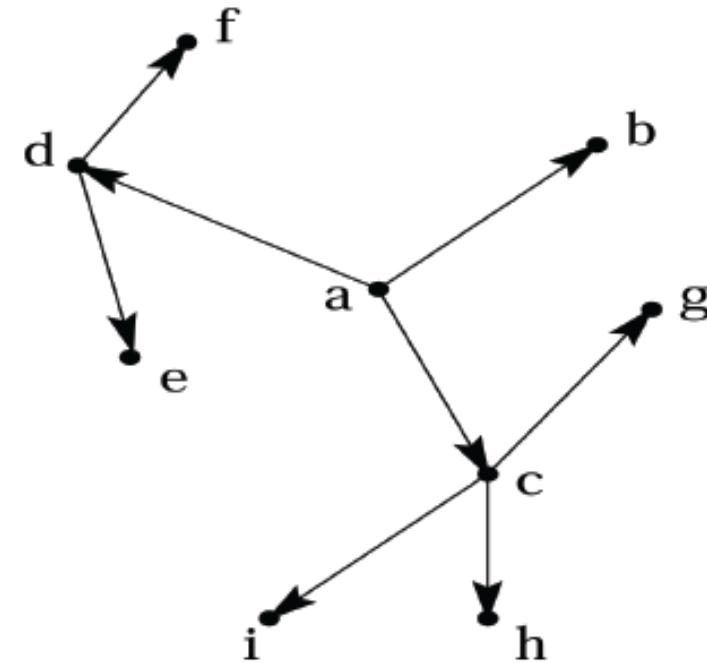
grafo non diretto, connesso e aciclico

Sia $R \subseteq S \times S$ dove $S = \{a, b, c, d, e, f, g, h, i\}$

$$R = \{\langle a, b \rangle, \langle a, c \rangle, \langle a, d \rangle, \langle c, g \rangle, \langle c, h \rangle, \langle c, i \rangle, \langle d, e \rangle, \langle d, f \rangle, \langle h, g \rangle\}.$$



ALBERI



Esempio di grafo che ha struttura di albero

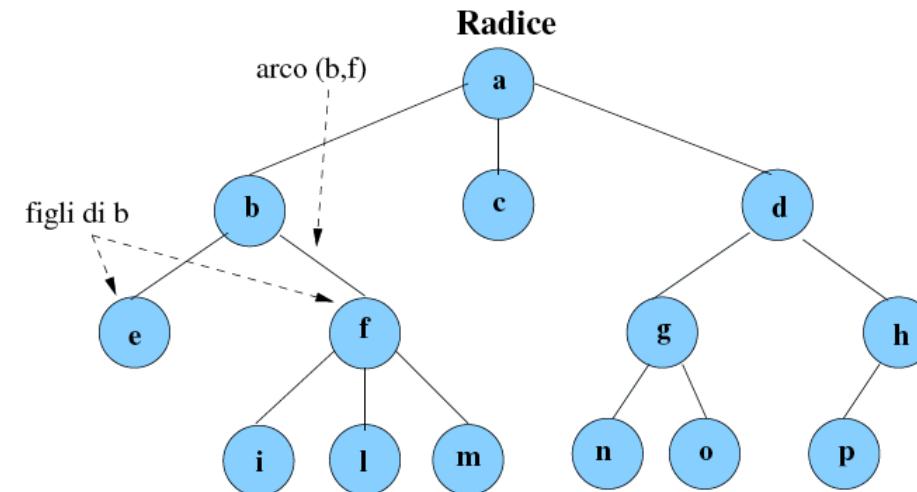
CAMMINO SU UN ALBERO

Un **cammino dal nodo i al nodo j** è la sequenza di archi che devono essere attraversati per raggiungere il nodo j partendo dal nodo i

Ogni nodo y che si trova sul cammino tra r e x è un **ascendente** di x ; viceversa, x è un **descendente** di y

- r è l'unico nodo che non ha ascendenti

Se l'ultimo arco del cammino da r a x è (y, x) , allora y è il **padre** di x , e x è **figlio** di y

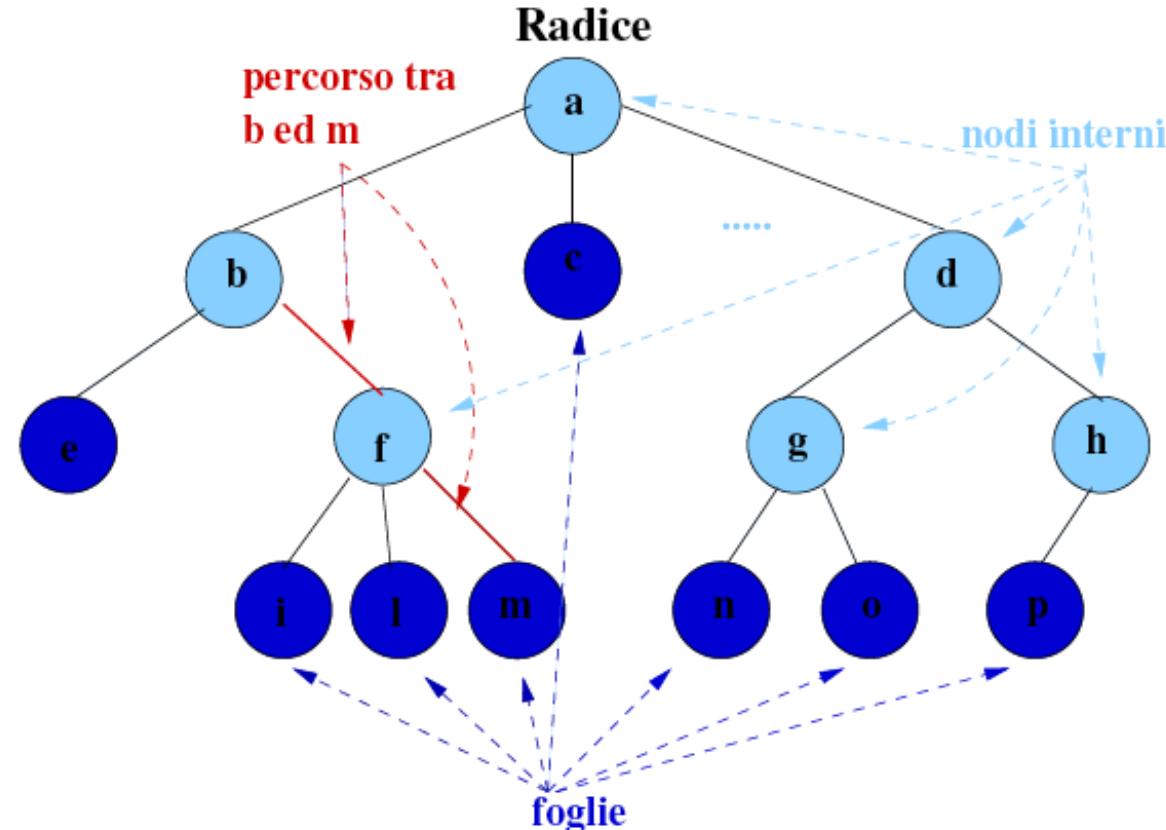


Il numero di figli di un nodo x si dice **grado** di x

FOGLIE, NODI INTERNI, PERCORSI

Un nodo che non ha figli si dice **nodo foglia**

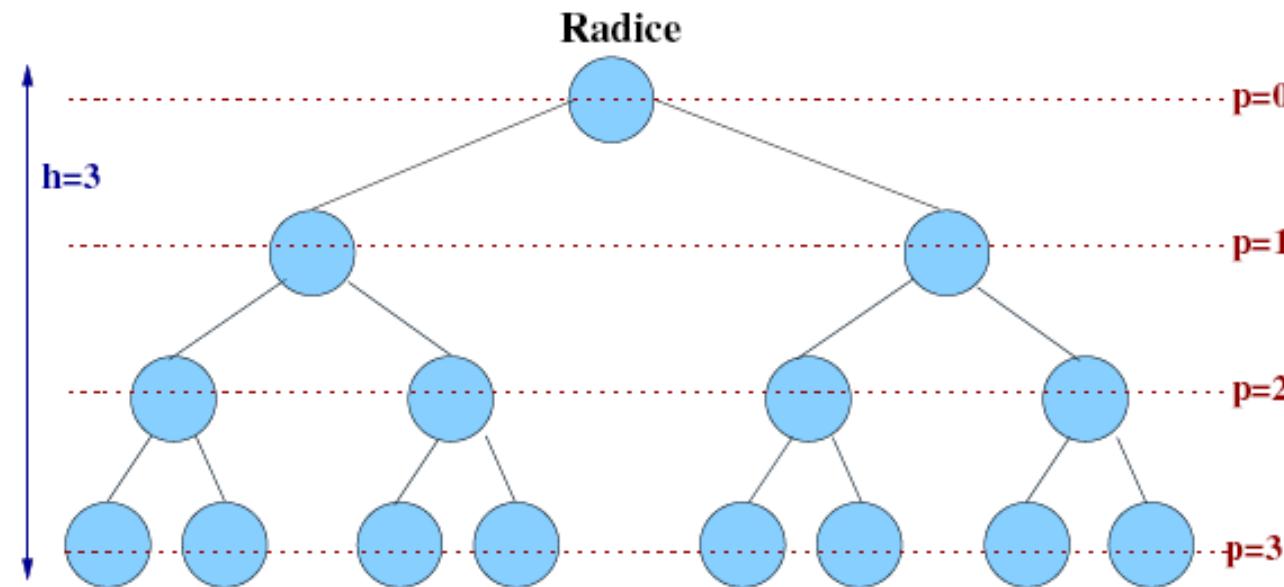
Un nodo si dice **nodo interno** se ha almeno un figlio.



PROFONDITA' E ALTEZZA

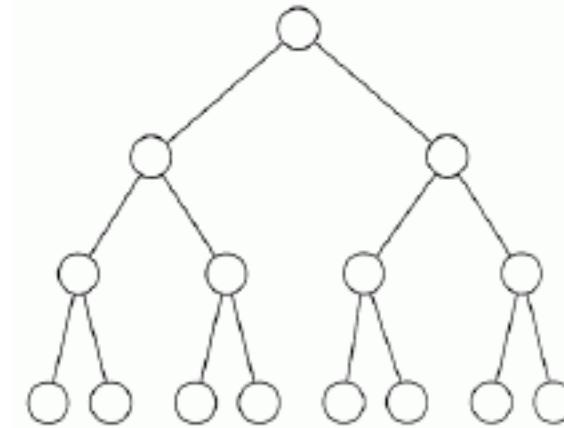
La **profondità di un nodo** x è la lunghezza del percorso per andare da r a x .

L'**altezza dell'albero** è la profondità massima che può avere un nodo dell'albero.



ALBERO BINARIO

Un *albero binario* è un albero in cui ogni nodo diverso dalle foglie ha al massimo due figli ordinati, detti *figlio sinistro* e *figlio destro*.

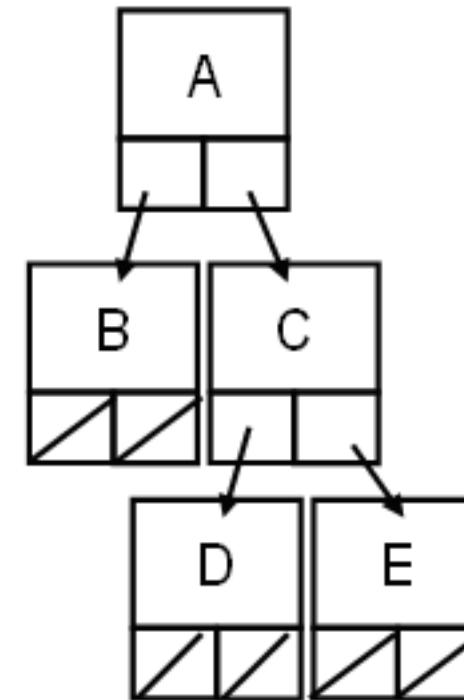


ALBERO BINARIO: PROPRIETA'

Un albero binario con N nodi interni ha $N+1$ nodi esterni.

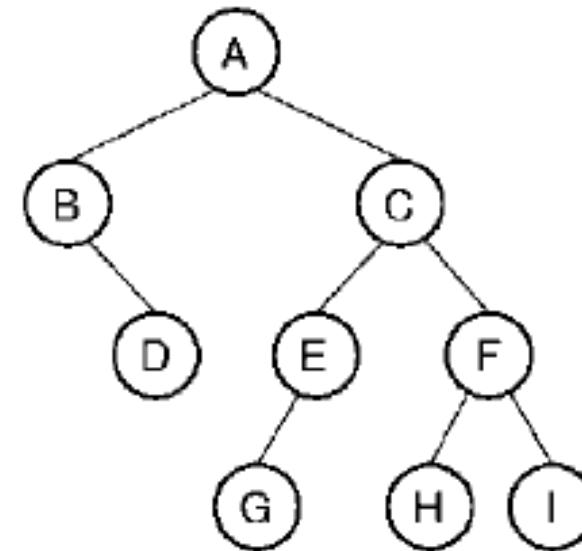
ALBERO BINARIO

1	A	1
2	B	1
3	C	1
4	0	0
5	0	0
6	D	1
7	E	1



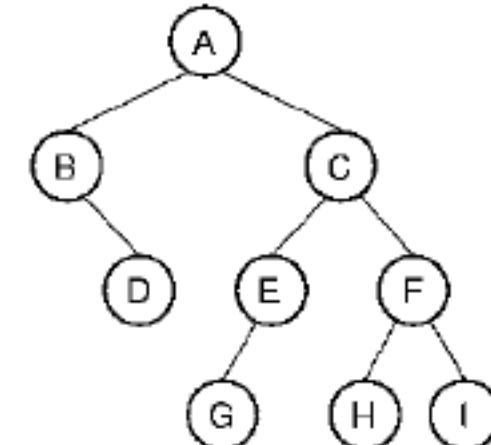
ALBERO BINARIO

E' una struttura costituita da un insieme finito di elementi detti nodi. Quest'insieme è vuoto oppure è costituito da una radice e da due alberi binari (sottoalbero sinistro e destro) disgiunti tra loro e dalla radice.



ALBERO BINARIO

- Un nodo senza figli si dice **foglia** o **nodo esterno**
- Un nodo con almeno un figlio si dice **nodo interno**
- Un albero binario si dice **pieno** (**full**) se ogni nodo è o un nodo interno con due figli non vuoti oppure una foglia
- Un albero binario si dice **completo** se ha altezza d e tutti i primi $d-1$ livelli sono occupati da nodi. Tutti i nodi del livello d stanno a



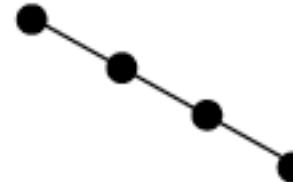
(a)



(b)

TEOREMA DELL'ALBERO BINARIO PIENO

- Per analizzare lo spazio occupato: rapporto minimo e massimo di nodi foglia rispetto ai nodi interni
 - numero minimo di nodi foglia in un albero: 1



- numero massimo di nodi foglia in un albero: si ha quando ogni nodo interno ha due nodi figli non vuoti --> albero pieno

TEOREMA DELL'ALBERO BINARIO PIENO

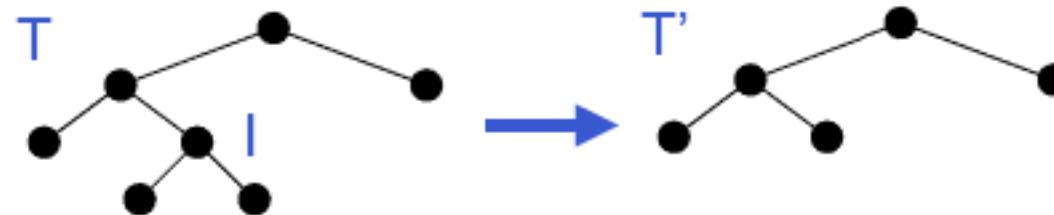
- Teorema: *il numero di foglie in un albero binario (non vuoto) pieno è pari al numero di nodi interni + 1.*

- Dimostrazione per induzione:

Caso base: Un albero binario pieno con 1 nodo interno deve avere due nodi figli.

Ipotesi di induzione: Un albero binario T con $n-1$ nodi interni ha n foglie.

Passo di induzione: Dato un albero T con n nodi interni, si consideri il nodo interno I con due figli foglia. Sia T' l'albero ottenuto rimuovendo i figli di I . Per l'ipotesi di induzione T' è un albero binario pieno con n foglie. Aggiungendo a T' i figli di I il numero di nodi interni aumenta di 1 e diventa n . Anche il numero di figli aumenta di 1.



TEOREMA DELL'ALBERO BINARIO PIENO

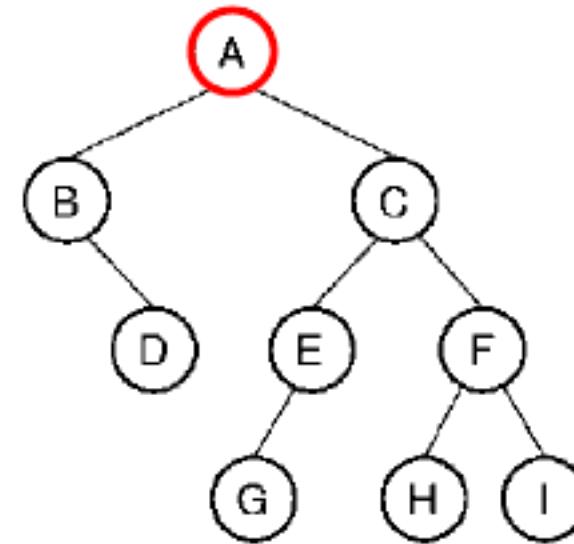
- Significato: in un albero binario pieno se il numero totale di nodi è N , il numero di nodi foglia è pari a $N/2 + 1$.
- Corollario: *Il numero di puntatori nulli in un albero binario non vuoto qualunque è pari al numero di nodi nell'albero + 1.*
- Dimostrazione:
si sostituiscano tutti i puntatori nulli con un puntatore ad una foglia vuota. Si ottiene un albero binario pieno.

ATTRaversamento di un ALBERO BINARIO

- Attraversamento di un albero: visita di tutti i suoi nodi secondo un particolare ordine
 - un attraversamento che elenca tutti i nodi esattamente una volta si chiama enumerazione dei nodi dell'albero
- Attraversamenti
 - in profondità
 - in preordine si visitano i nodi prima di visitare i loro figli VLR
 - in postordine si visitano i nodi dopo aver visitato i loro figli LRV
 - “in ordine” prima si visita un figlio, poi il nodo, poi l’altro figlio LVR
 - in ampiezza

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

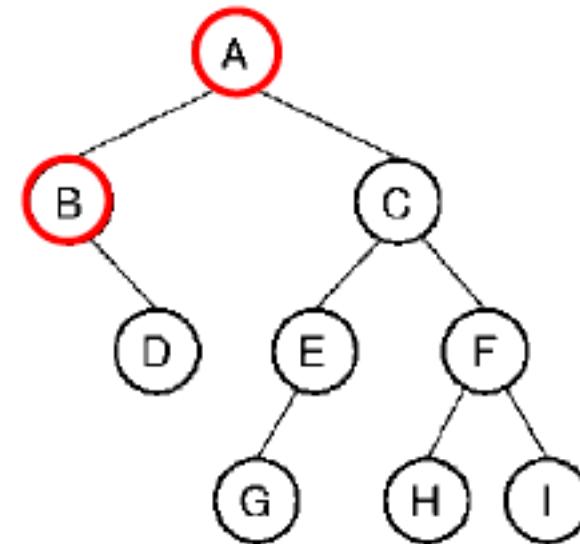
- VLR



ABDCEGFHI

ATTRaversamento di un albero binario PREORDER

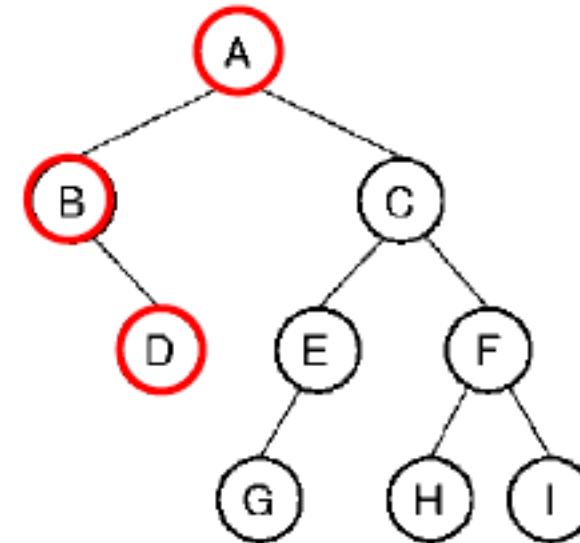
- VLR



ABDCEGFHI

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

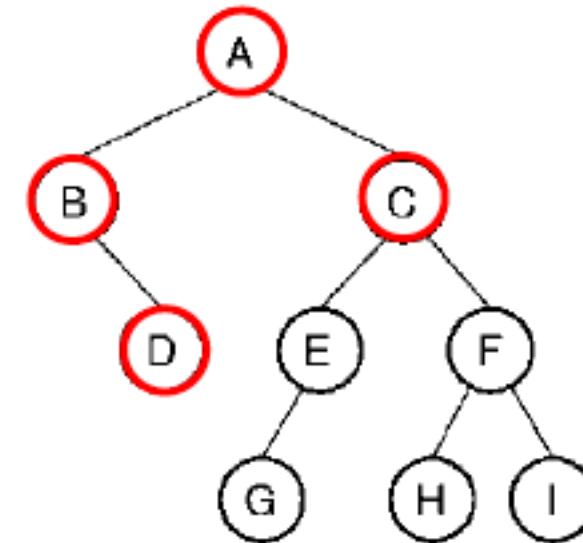
- VLR



ABDCEGFHI

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

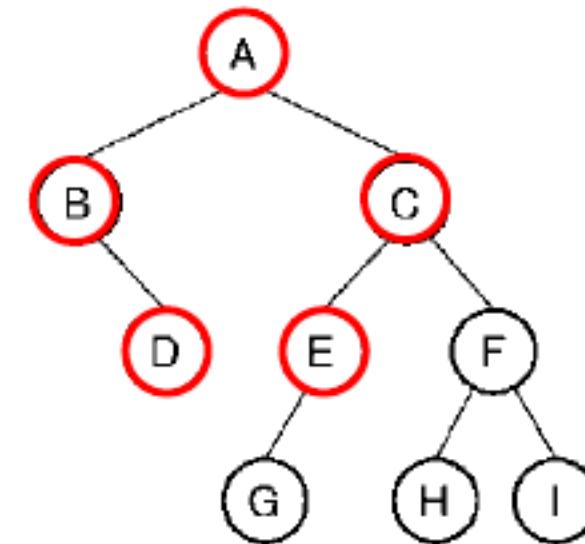
- VLR



ABD**C**E**G**F**H**I

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

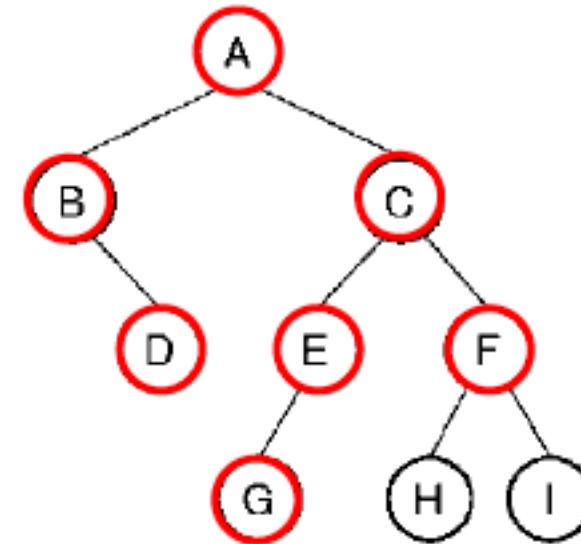
- VLR



ABDCEGFHI

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

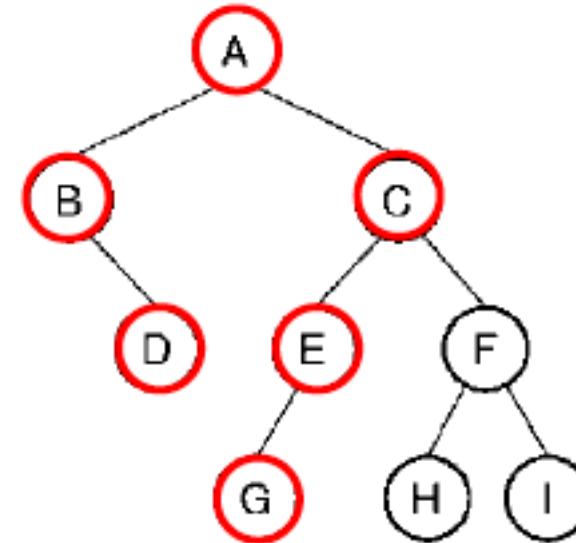
- VLR



ABDCEGFHI

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

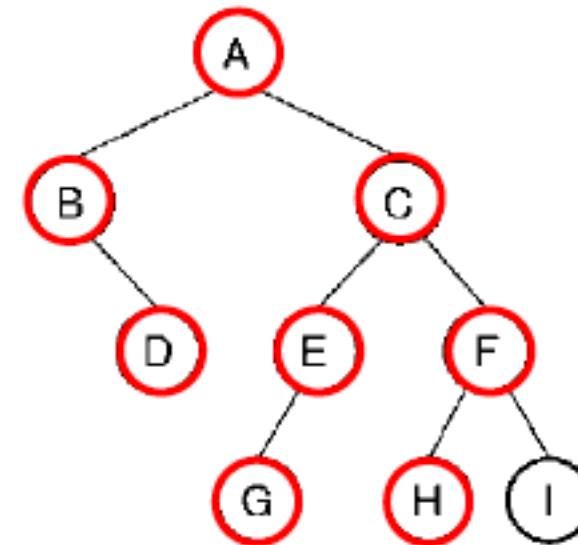
- VLR



ABDCEGFHI

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

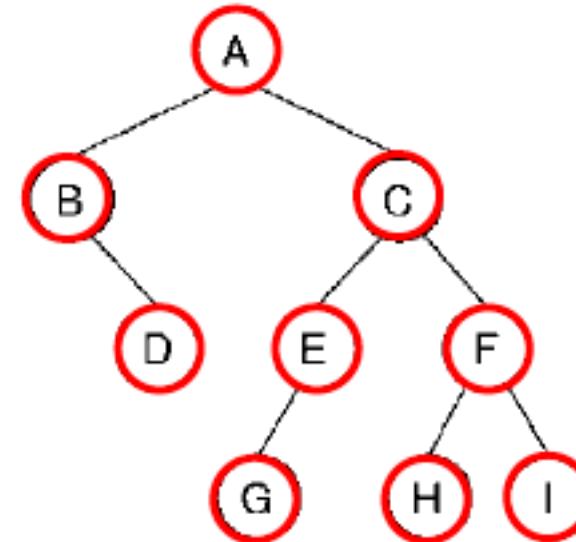
- VLR



ABDCEGFHI

ATTRAVERSAMENTO DI UN ALBERO BINARIO PREORDER

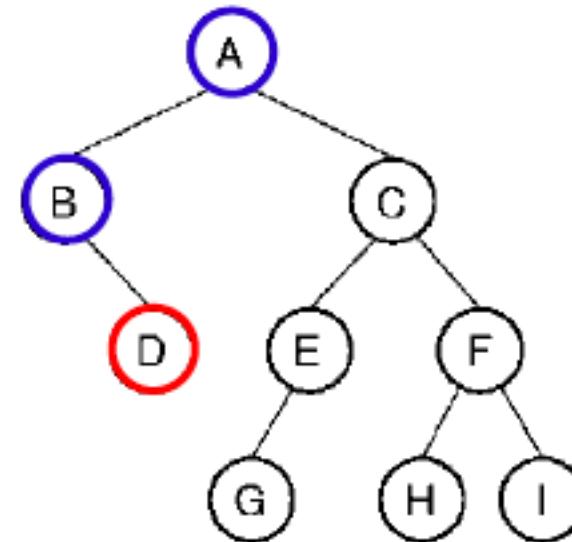
- VLR



ABDCEGFHI

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

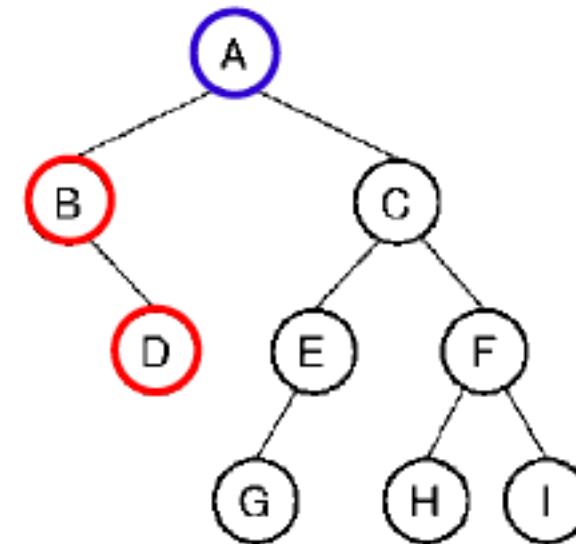
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

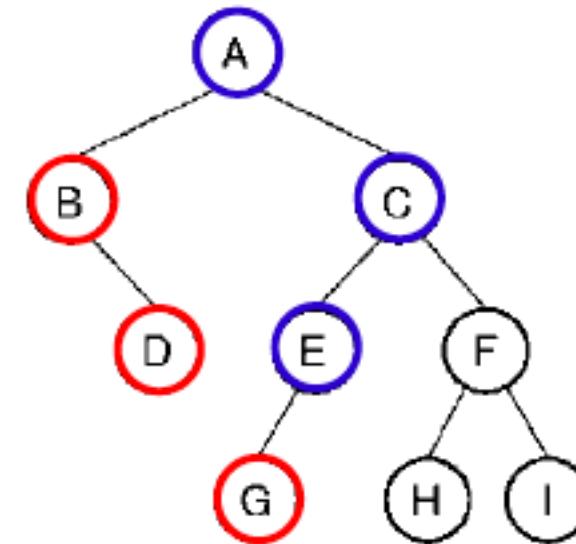
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

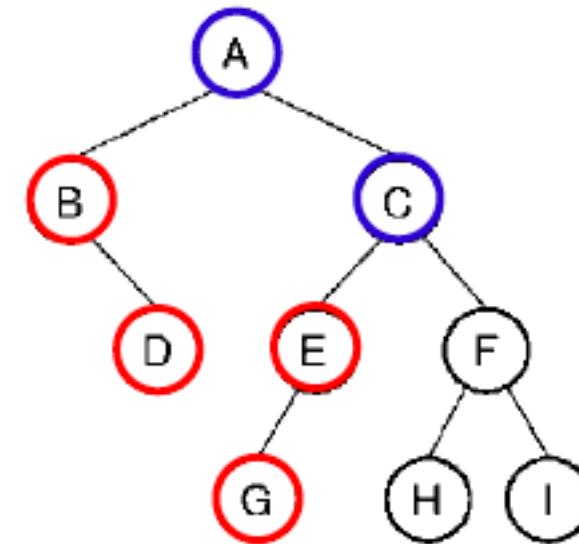
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

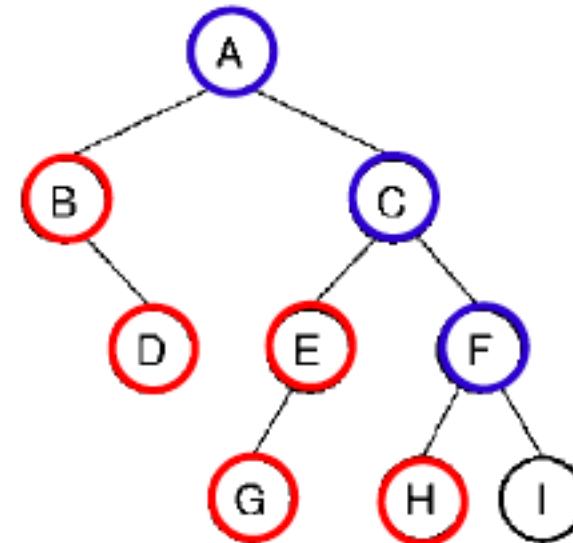
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

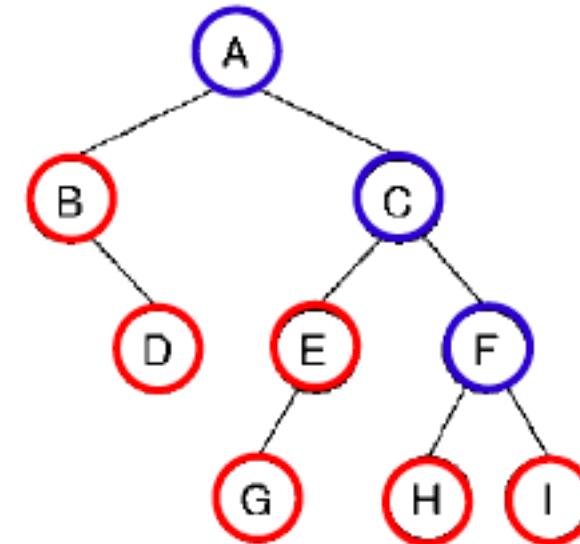
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

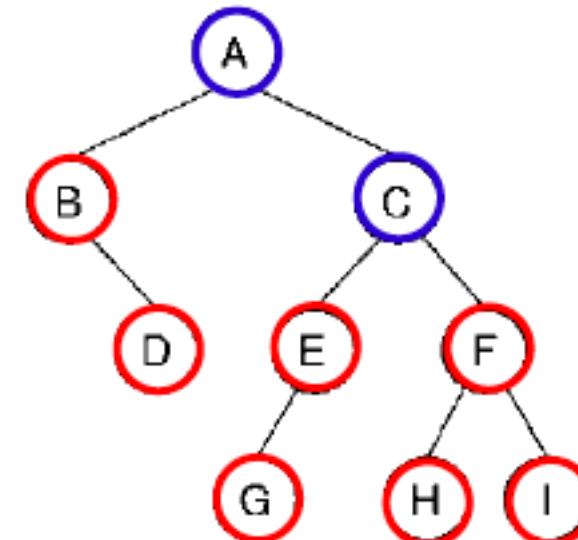
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

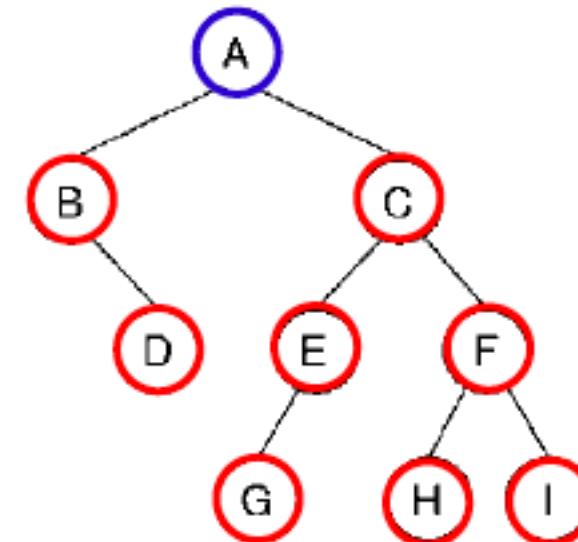
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

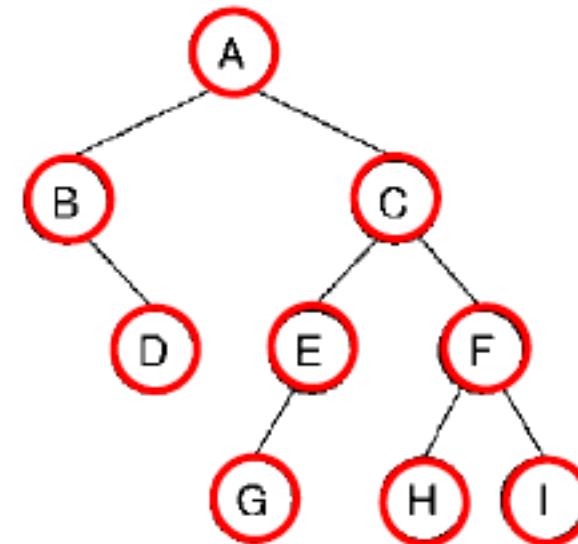
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO POSTORDER

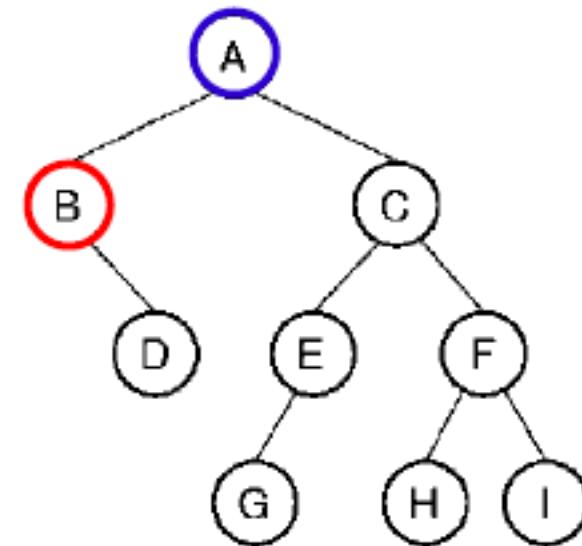
- LRV



DBGEHIFCA

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

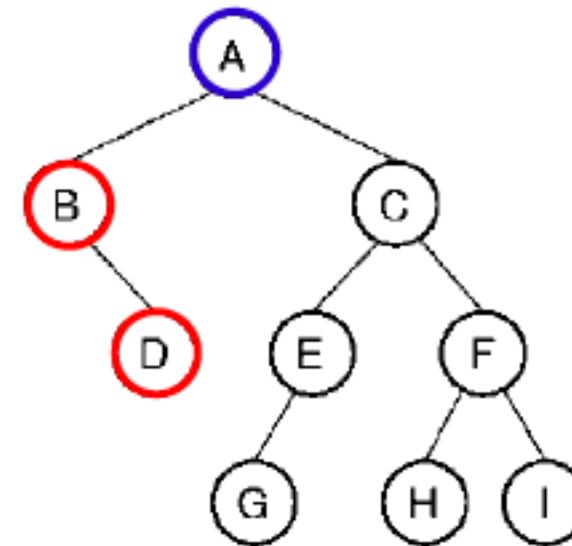
- LVR



BDAGECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

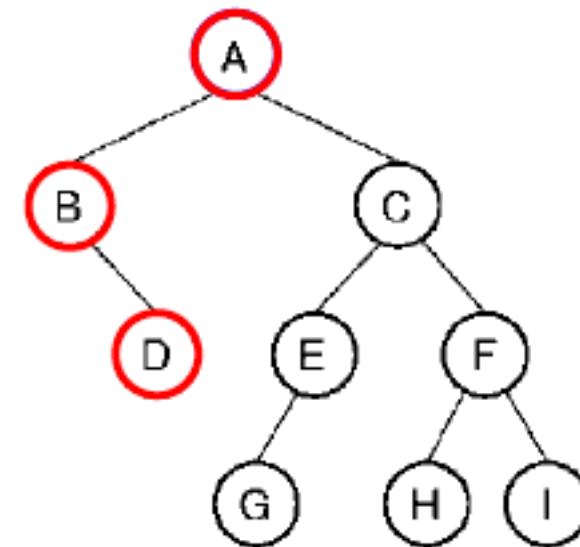
- LVR



BDAGECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

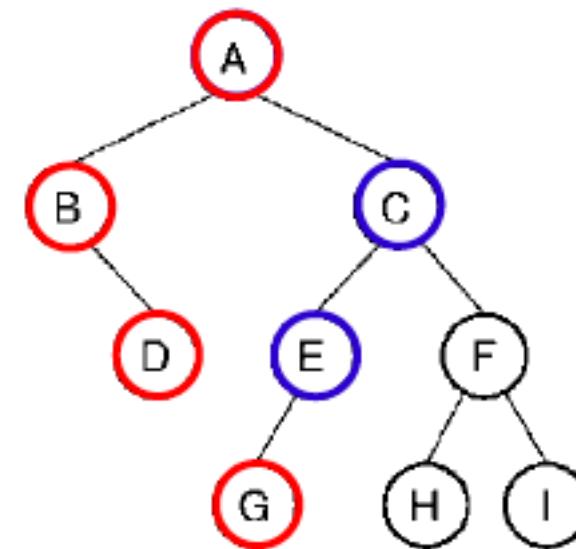
- LVR



BDA_{RED}GECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

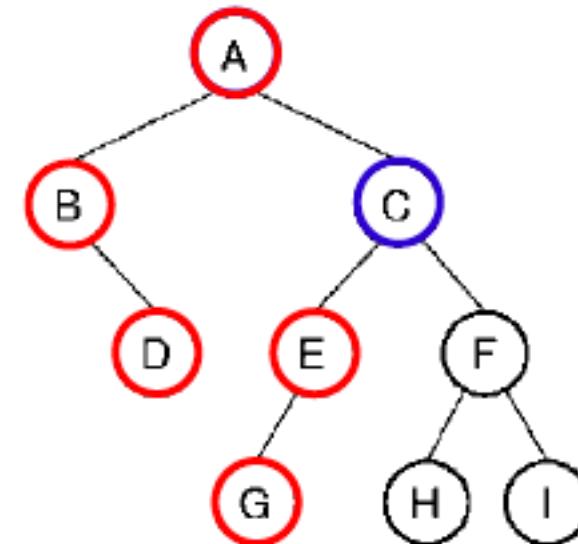
- LVR



BDAGECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

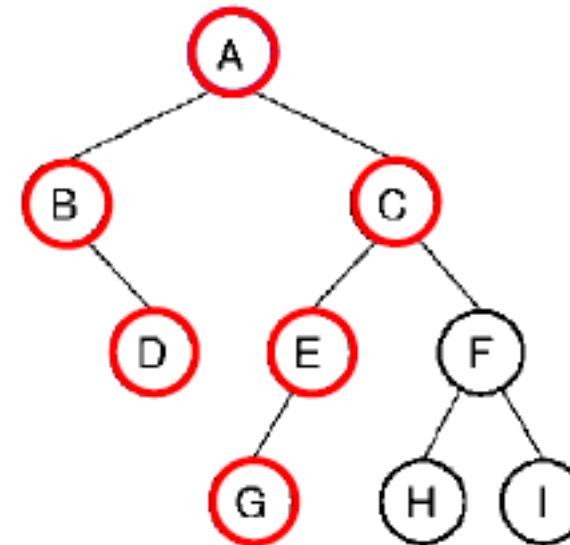
- LVR



BDAGECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

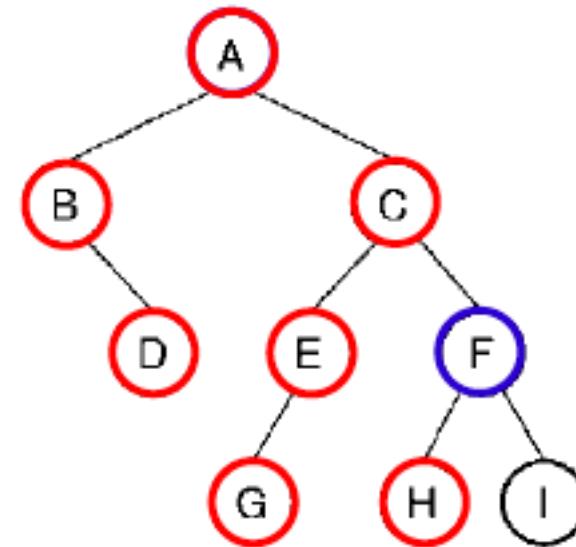
- LVR



BDAGECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

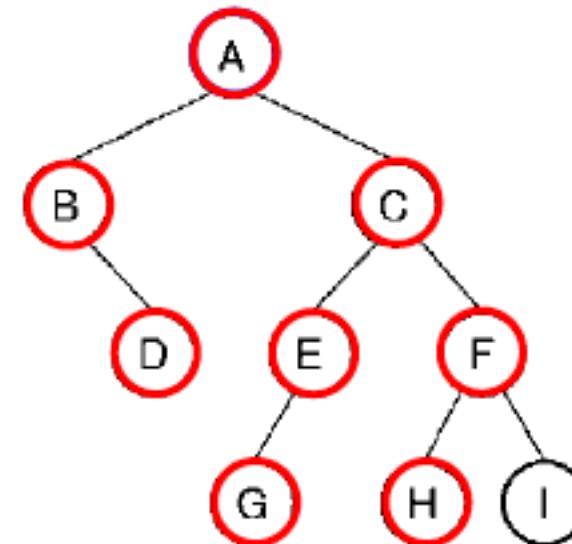
- LVR



BDAGECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

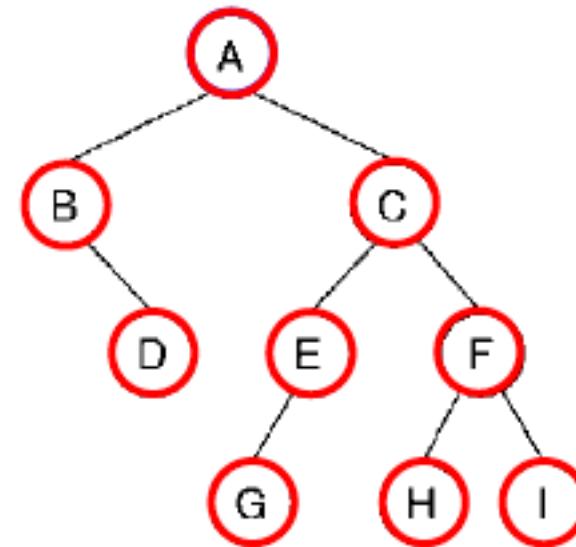
- LVR



BDAGECHFI

ATTRAVERSAMENTO DI UN ALBERO BINARIO ORDER

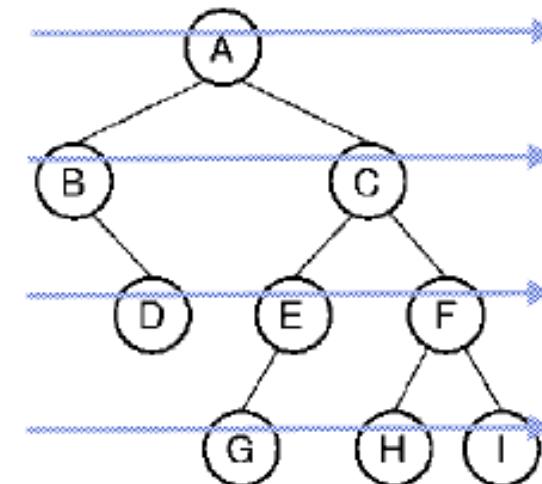
- LVR



BDAGECHFI

ATTRaversamento in ampiezza

- Attraversamento in ampiezza (breadth-first): visita di ciascun nodo per livelli



ABCDEFGHI

- La realizzazione è semplice se si utilizza una coda

FONDAMENTI DELL'INFORMATICA

STRUTTURE RELAZIONALI, GRAFI E ORDINAMENTI (parte 4)

END