



UNIVERSITÀ
DEGLI STUDI
DELL'AQUILA



DISIM
Dipartimento di Ingegneria
e Scienze dell'Informazione
e Matematica



Laboratorio di Algoritmi e Strutture Dati a.a. 2024/2025

Alberi di ricerca bilanciati

Alberi 2-3-4

Alberi red-black

Giovanna Melideo

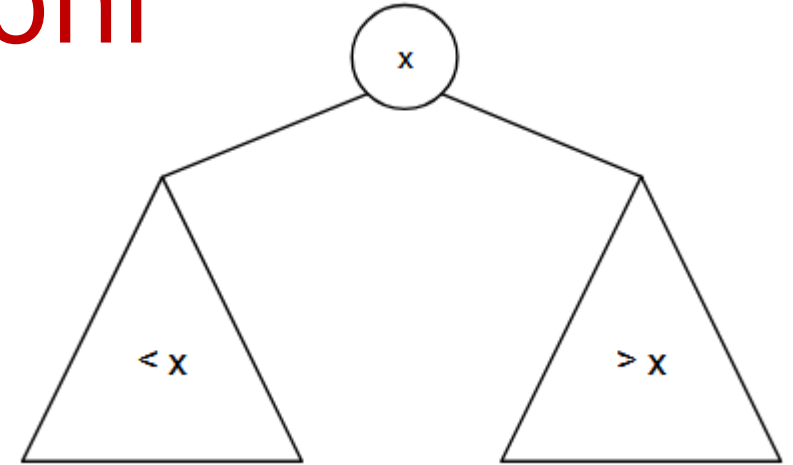
Università degli Studi dell'Aquila
DISIM

Alberi Binari di Ricerca (richiami)

- Un albero binario di ricerca (ABR, BST) è un albero nel quale vengono mantenute le seguenti proprietà:
 - ogni nodo contiene una chiave
 - il valore della chiave contenuta in ogni nodo è maggiore della chiave contenuta in ciascun nodo del suo sottoalbero sinistro (se esiste)
 - il valore della chiave contenuta in ogni nodo è minore della chiave contenuta in ciascun nodo del suo sottoalbero destro (se esiste)
- I BST sono strutture dati che supportano tutte le operazioni:
 - inserimento, ricerca e cancellazione

BST: prestazioni

- Prestazioni variabili da:
 - logaritmiche: caso migliore, albero bilanciato
 - lineari: caso peggiore, albero degenero.
- Soluzioni:
 - applicazione periodica di procedure di ribilanciamento
 - imposizione di vincoli sull'albero per limitare lo sbilanciamento
 - Esempio: alberi AVL (fattore di bilanciamento)



Alberi di ricerca 2-3-4

- Alberi di ricerca (**non binari**), aventi 3 tipi di nodi:
- **2-nodi**: 1 chiave, sottoalbero sinistro delle chiavi minori, sottoalbero destro delle chiavi maggiori
- **3-nodi**: 2 chiavi ordinate, sottoalbero sinistro delle chiavi minori di entrambe le chiavi, sottoalbero centrale delle chiavi comprese tra le due, sottoalbero destro delle chiavi maggiori di entrambe le chiavi
- **4-nodi**: 3 chiavi ordinate, 4 sottoalberi con chiavi con valori che stanno negli intervalli di valori definiti dalle 3 chiavi.

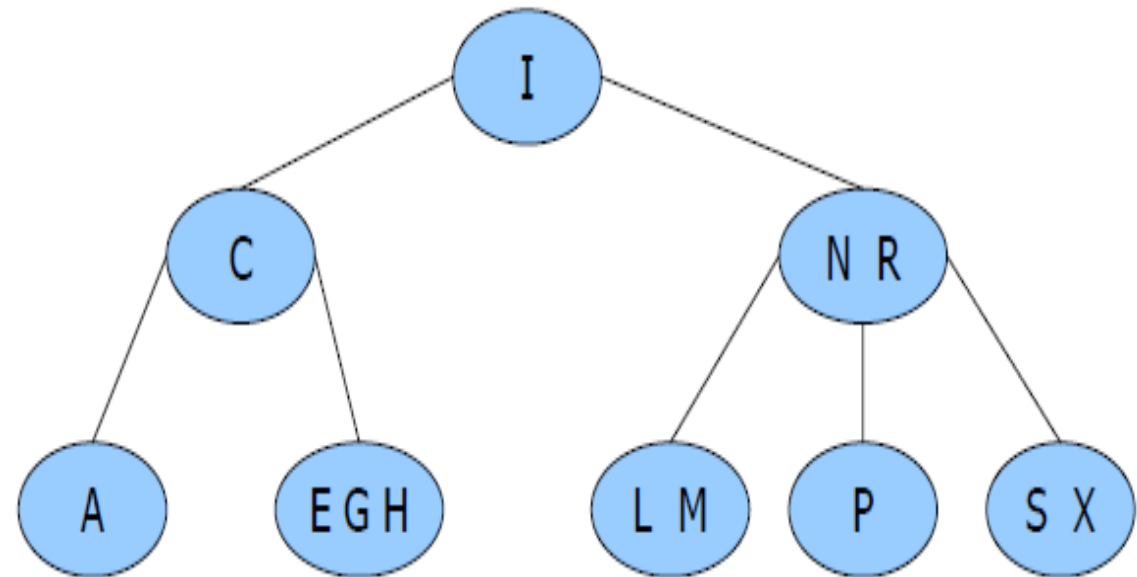
Alberi di ricerca bilanciati 2-3-4

- **Bilanciamento:** tutte le foglie hanno uguale distanza dalla radice
 - Si noti che un BST è un albero 2-3-4 non necessariamente bilanciato formato da soli 2-nodi

Alberi di ricerca bilanciati 2-3-4: esempio

Sono soddisfatte le proprietà di:

- Ordinamento delle chiavi
- Bilanciamento

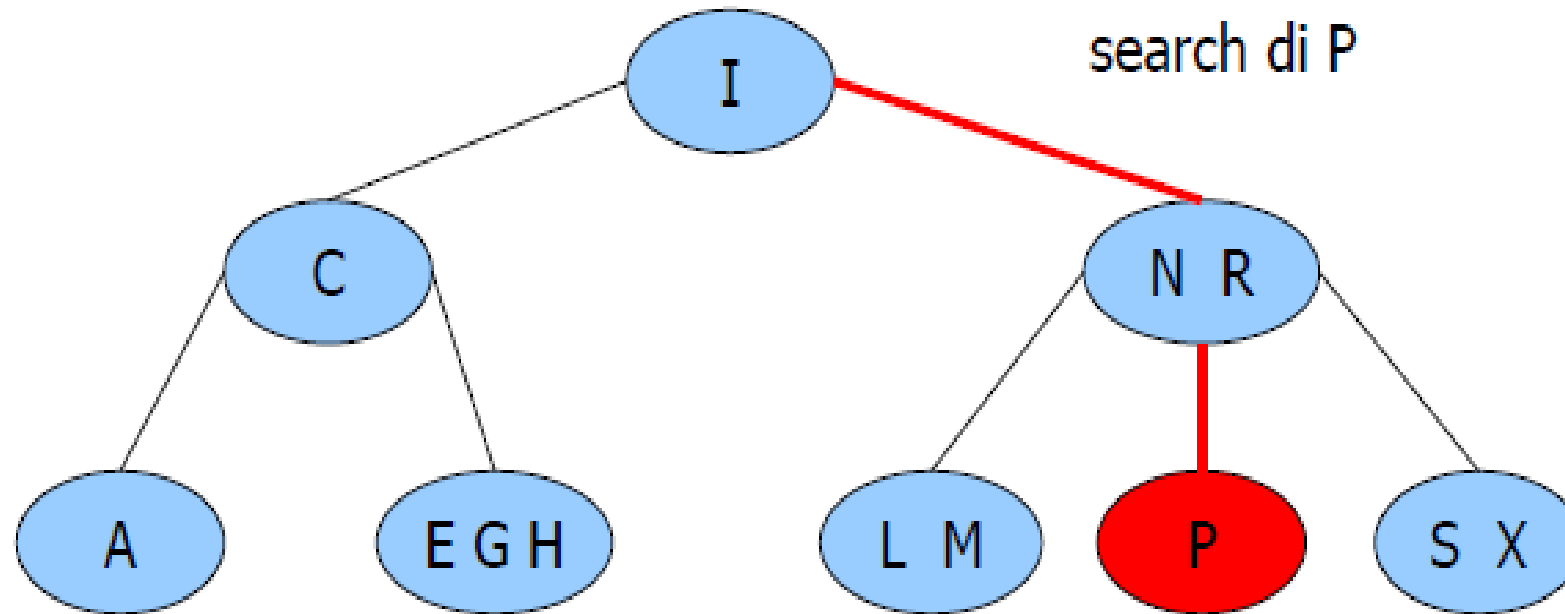


Ricerca di una chiave

Generalizzazione della ricerca nei BST:

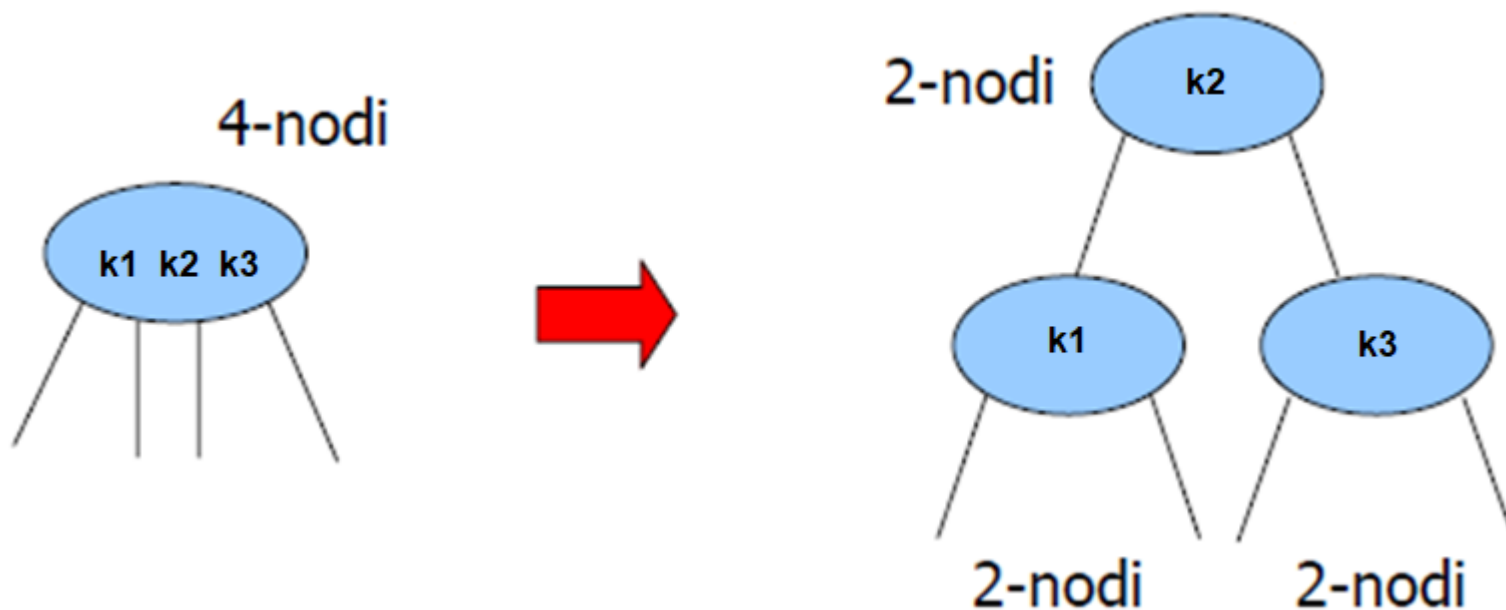
- Confronto sequenziale della chiave cercata con le chiavi contenute nella radice
- search hit se trovata
- se non trovata, si scende nel sottoalbero che corrisponde all'intervallo di valori che comprende la chiave
- Si ripete (ricorsivamente) la ricerca nel sottoalbero.

Ricerca di una chiave: esempio



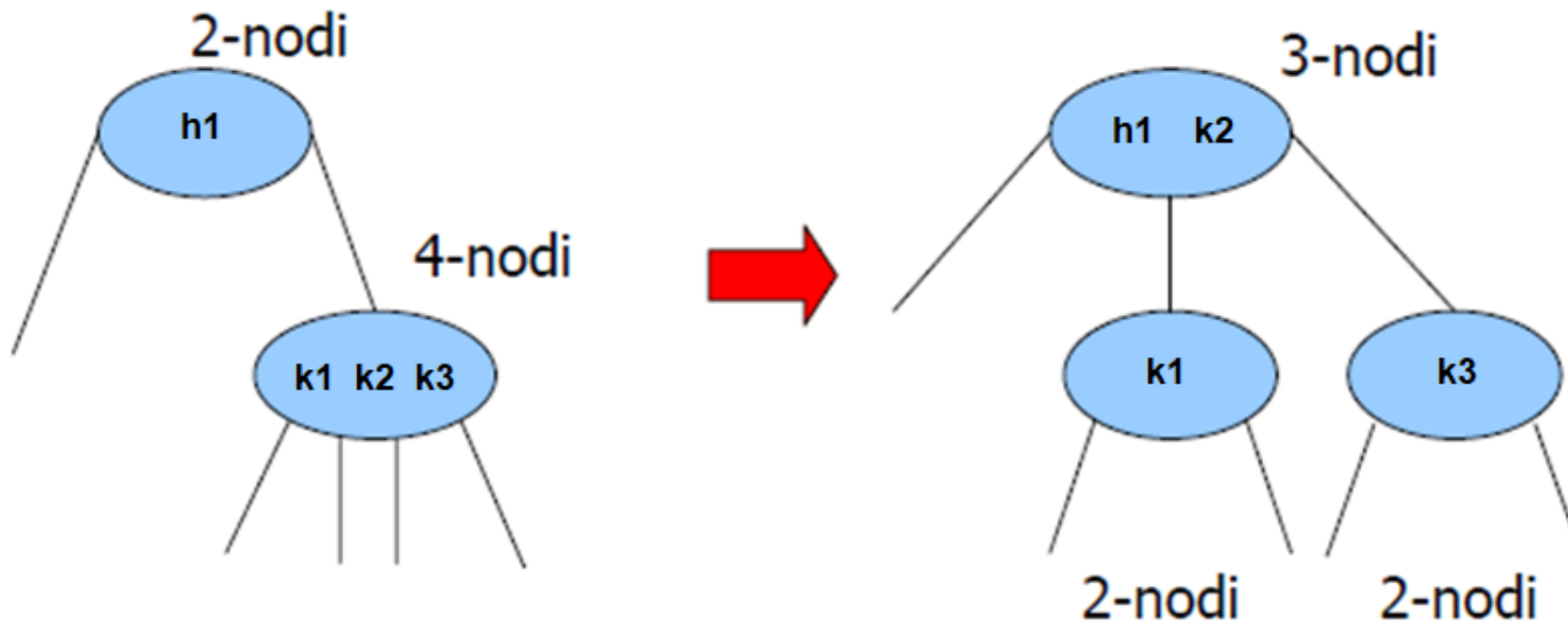
Split di un 4-nodo radice

- Si trasforma un 4-nodo in tre 2-nodi
- L'altezza dell'albero cresce di 1
- L'albero cresce sempre dalla radice verso l'alto



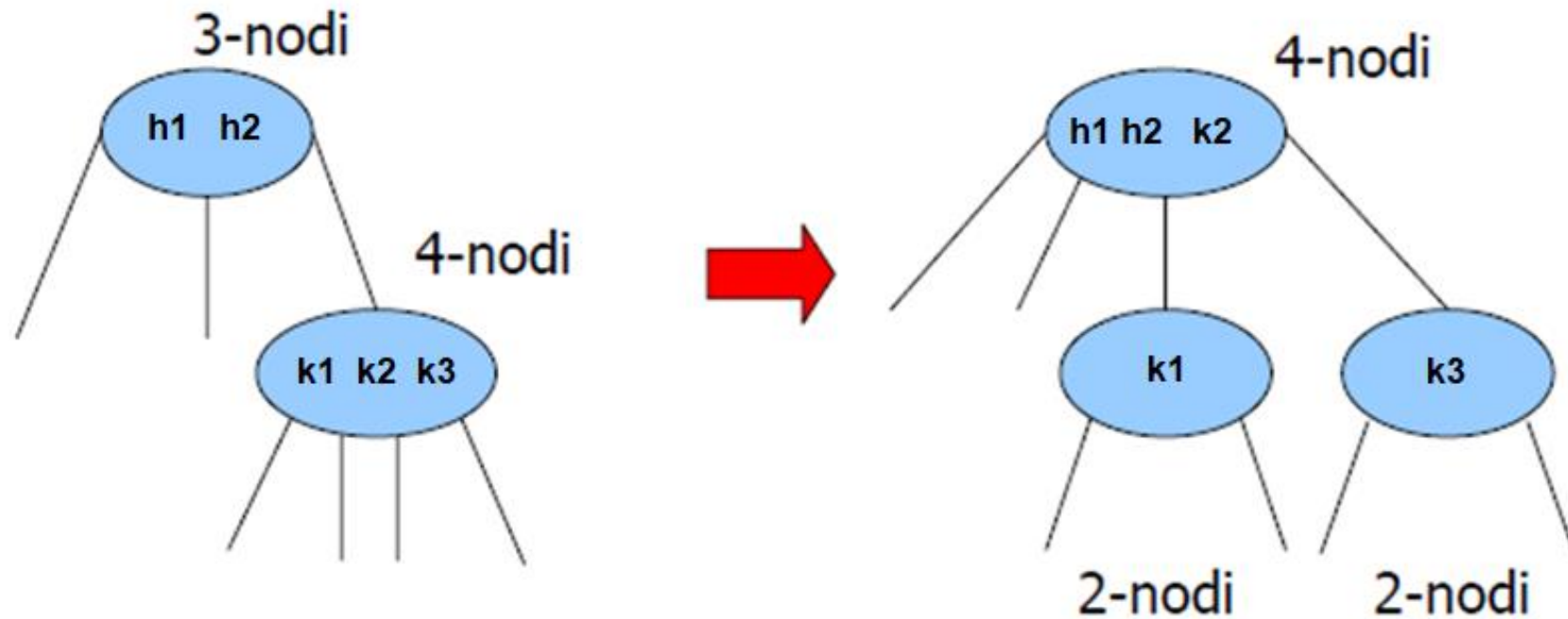
Split di un 4-nodo figlio di un 2-nodo

- Si trasforma il 2-nodo padre in un 3-nodo inserendo in modo ordinato la chiave centrale del 4-nodo figlio



Split di un 4-nodo figlio di un 3-nodo

- Si trasforma il 3-nodo padre in un 4-nodo inserendo in modo ordinato la chiave centrale del 4-nodo figlio



Inserimento bottom-up

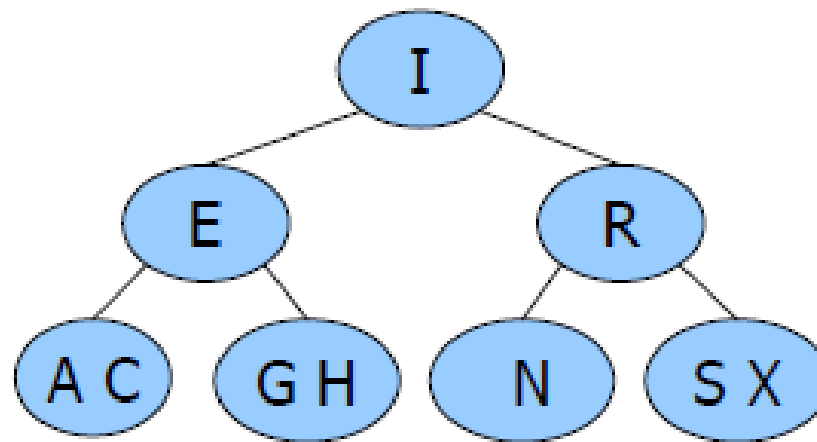
- Generalizzazione dell'inserimento nei BST:
 - si ricerca la nuova chiave a partire dalla radice
 - se la chiave è contenuta nell'albero l'inserimento fallisce, altrimenti la ricerca termina su una foglia:
 - se la foglia è un 2-nodo o 3-nodo, si inserisce ordinatamente la chiave trasformando rispettivamente la foglia in 3-nodo o 4 nodo;
 - se la foglia è un 4-nodo figlio di un 2-nodo o 3-nodo si effettua uno split del nodo foglia;
 - se la foglia è un 4-nodo figlio di un 4-nodo, si effettua prima uno split del padre e poi uno split della foglia;
 - ma anche il padre del padre potrebbe essere un 4-nodo...

Inserimento top-down

- Durante la ricerca a partire dalla radice per identificare il nodo in cui inserire la nuova chiave:
 - se il nodo corrente (radice compresa) **sul cammino di ricerca dalla radice alla foglia** è un 4-nodo, si effettua uno split del nodo
 - si noti che il nodo padre del 4-nodo (se esiste) può essere solo un 2-nodo o un 3-nodo, per cui quando la ricerca termina su una foglia si procede con eventuali operazioni di split come per l'inserimento bottom-up.

Inserimento top-down: esempio

- Inserimento in sequenza dei caratteri A S E R C H I N G X



Analisi della complessità

- Una ricerca in un albero 2-3-4 bilanciato di N nodi non visita mai più di $\lg N + 1$ nodi
- Un inserimento in un albero 2-3-4 bilanciato di N nodi richiede nel caso peggiore al più $\lg N + 1$ split.

Alberi red-black (RBtree) – Def. 1

Definizione 1

BST in cui:

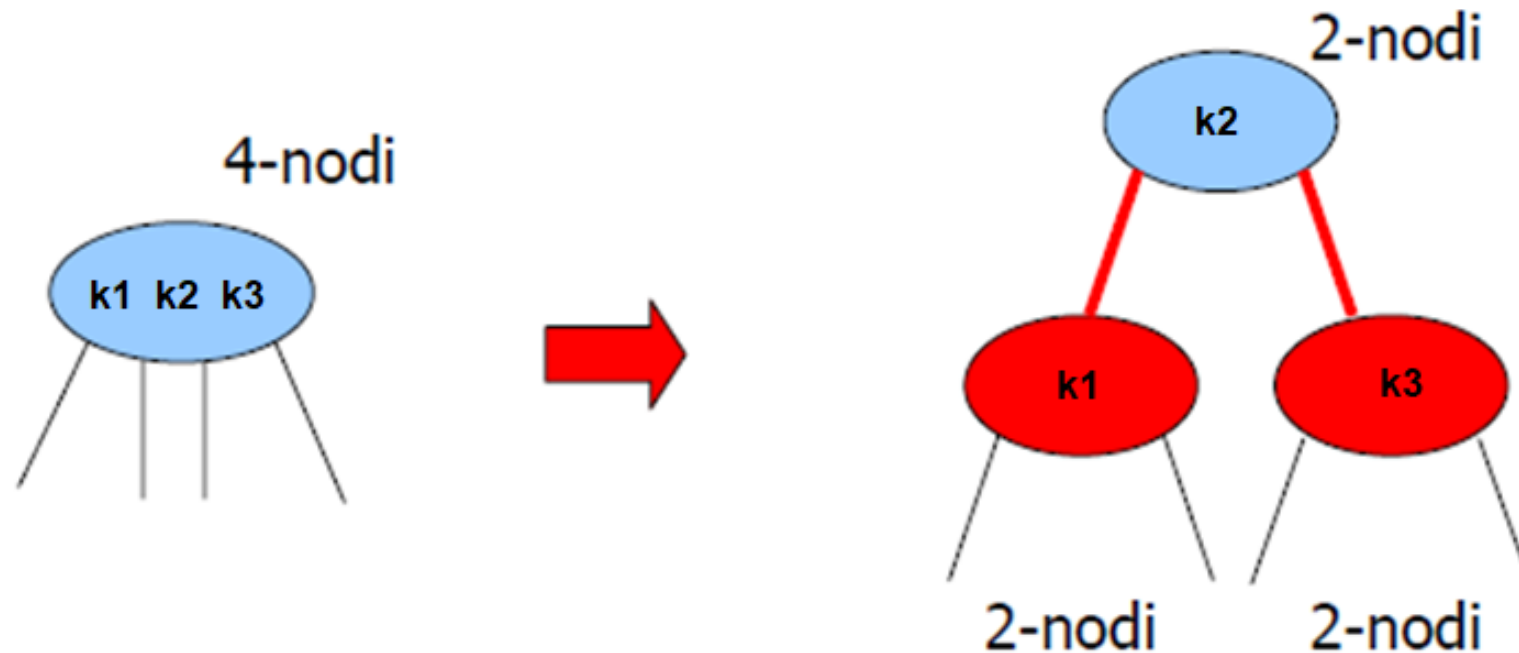
- ogni nodo è o rosso o nero
- se un nodo è rosso, non può avere figli rossi
- ogni cammino semplice dalla radice a una foglia contiene lo stesso numero di nodi neri

Alberi red-black (RBtree) – Def. 2

Definizione 2

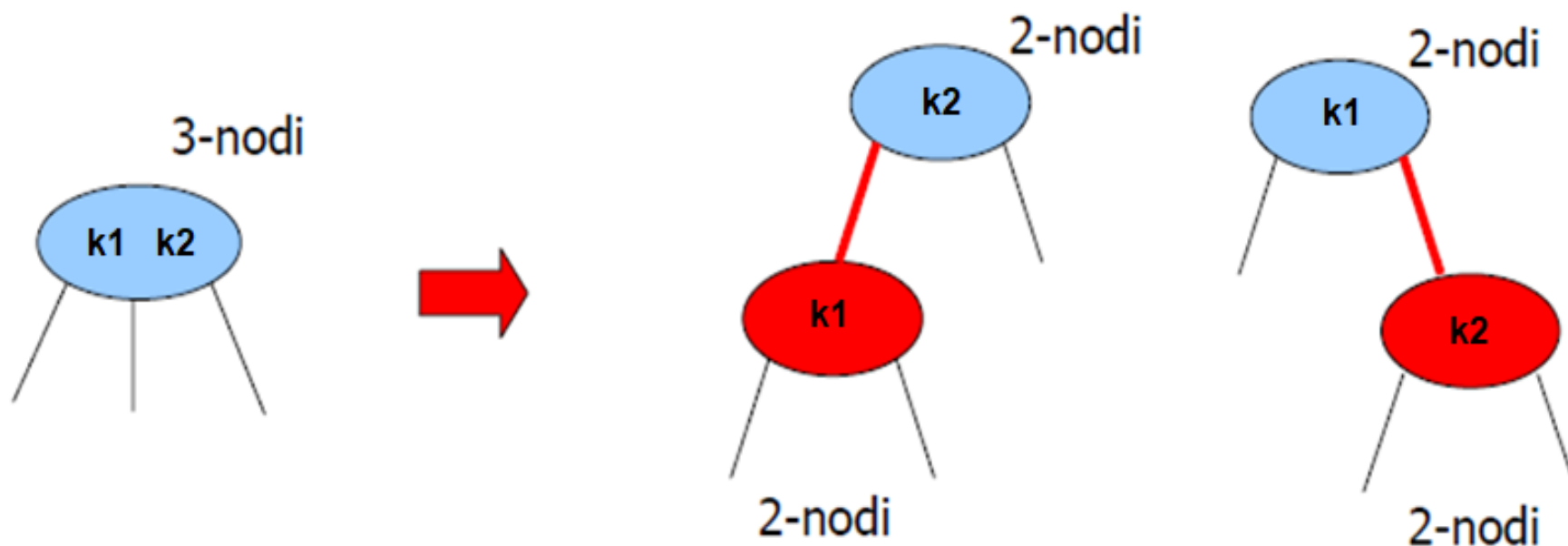
- Rappresentazione degli alberi 2-3-4 come BST con ulteriore bit di informazione per codificare 2-nodi e 3-nodi:
- **link rossi** che connettono piccoli alberi binari che formano 3-nodi e 4-nodi
- **link neri** che connettono l'intero albero 2-3-4
- Ogni nodo è raggiunto tramite 1 solo link, quindi colorare i link equivale a colorare i nodi.

Rappresentazione di un 4-nodo

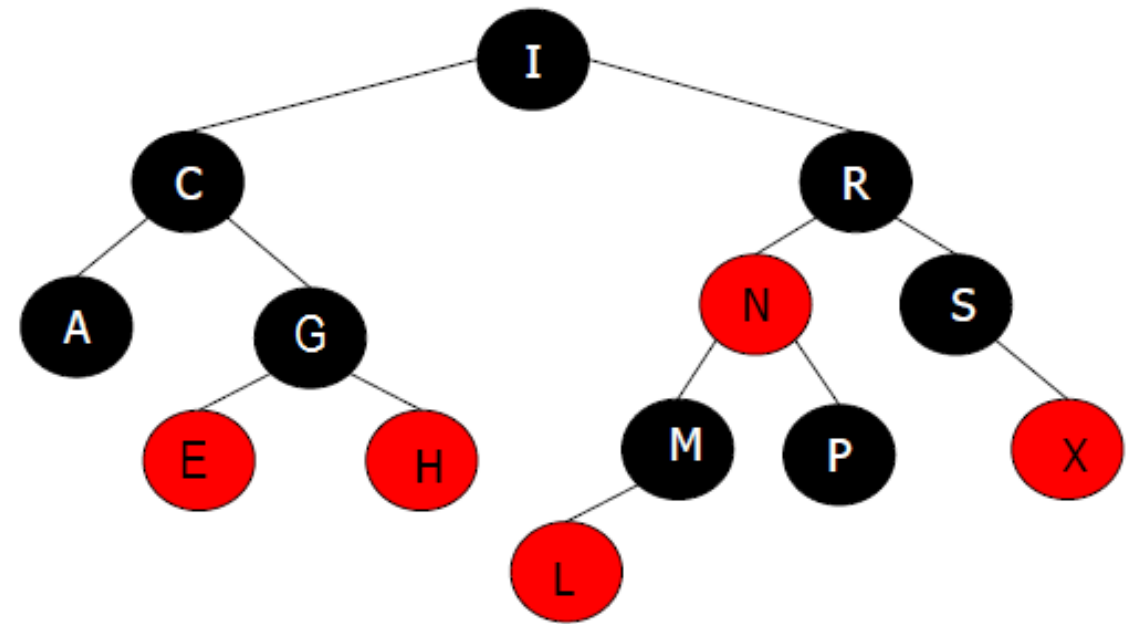
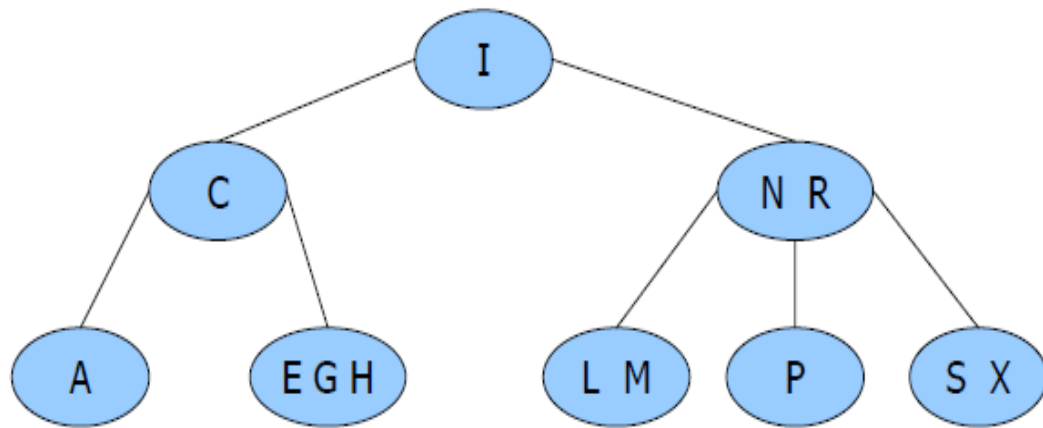


Rappresentazione di un 3-nodo

- Due alternative:



Dall'albero 2-3-4 al RBtree



Proprietà

- Gli alberi red-black corrispondono in modo diretto agli alberi 2-3-4
 - se eliminiamo i link rossi di un albero red-black e facciamo collassare i nodi connessi da tali link, otteniamo l'albero 2-3-4 corrispondente
- Possiamo implementare gli algoritmi per gli alberi 2-3-4 bilanciati mantenendo la corrispondenza con gli alberi red-black

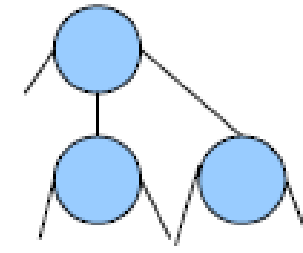
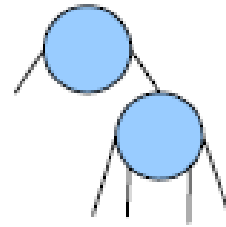
Algoritmi di ricerca e inserimento

- La procedura di **ricerca standard** su BST funziona sugli alberi red-black senza alcuna modifica
- La procedura di inserimento sugli alberi red-black è la diretta applicazione della procedura di **inserimento top-down** per gli alberi 2-3-4 bilanciati
 - immaginiamo di operare su un albero 2-3-4 implementato tramite un Rbtree

Inserimento top-down: split (1 di 5)

4-nodi figlio di 2-nodi (link DX)

Albero 2-3-4

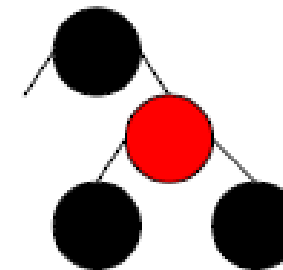
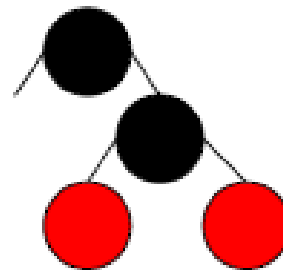


trasformazione



cambio di colore

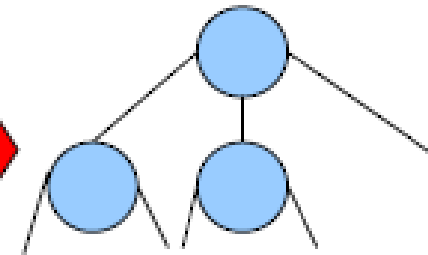
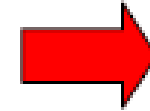
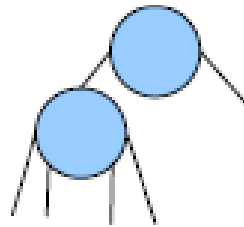
RB-tree



Inserimento top-down: split (2 di 5)

4-nodi figlio di 2-nodi (link SX)

Albero 2-3-4

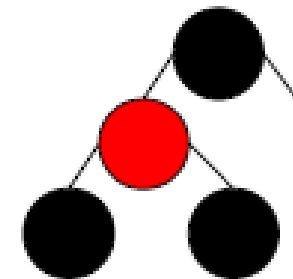
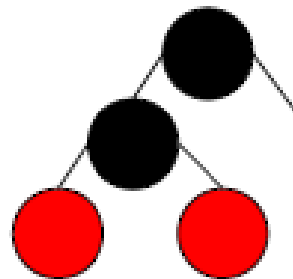


trasformazione



cambio di colore

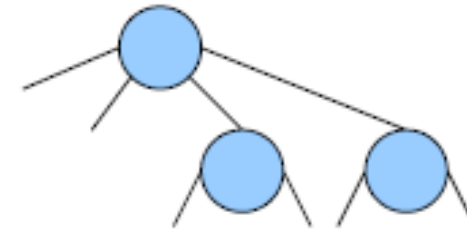
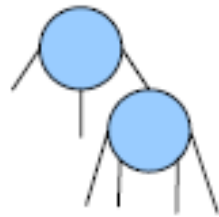
RB-tree



Inserimento top-down: split (3 di 5)

4-nodi figlio di 3-nodi (link DX)

Albero 2-3-4

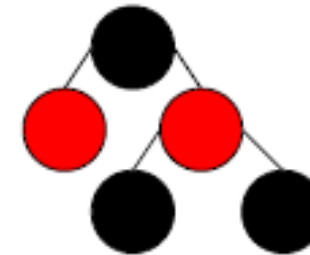
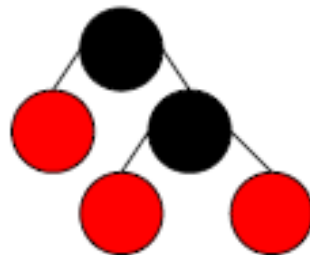


trasformazione



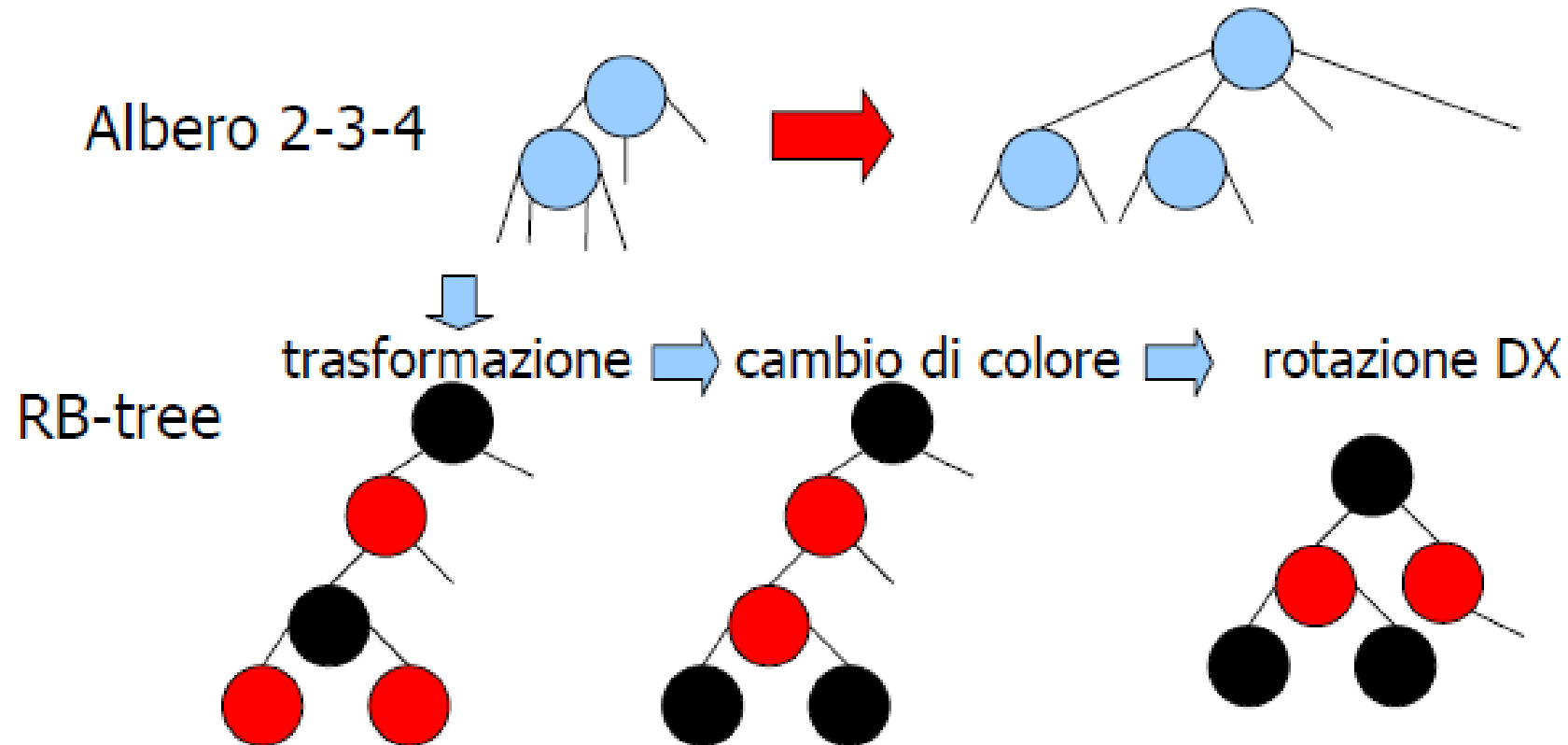
cambio di colore

RB-tree



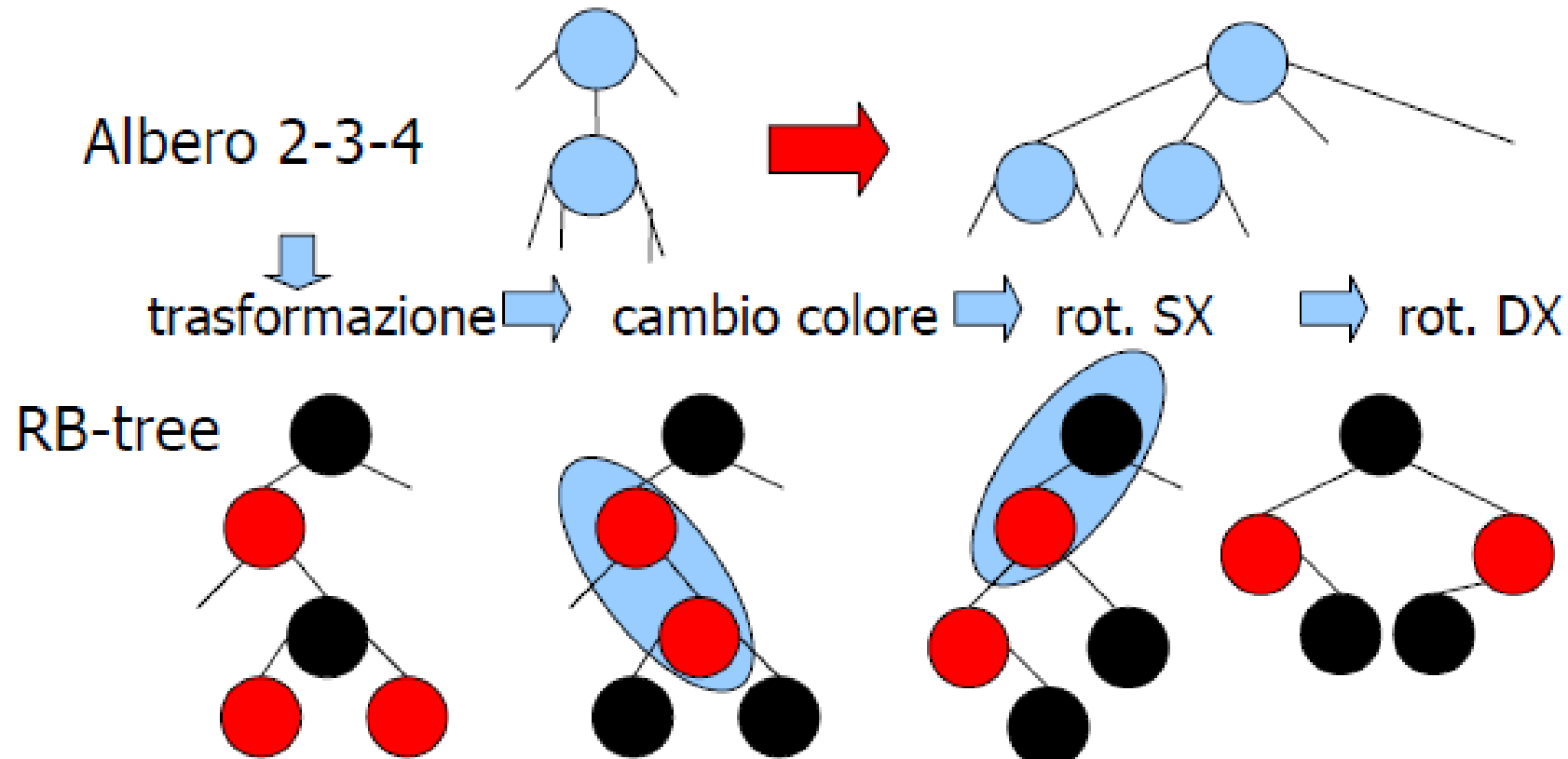
Inserimento top-down: split (4 di 5)

4-nodi figlio di 3-nodi (link SX)



Inserimento top-down: split (5 di 5)

4-nodi figlio di 3-nodi (link centrale)

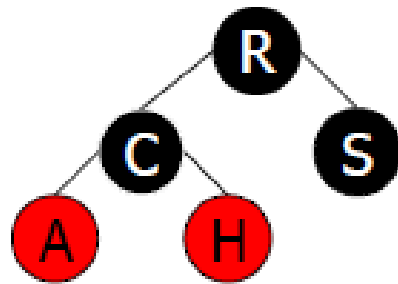


Algoritmo di inserimento

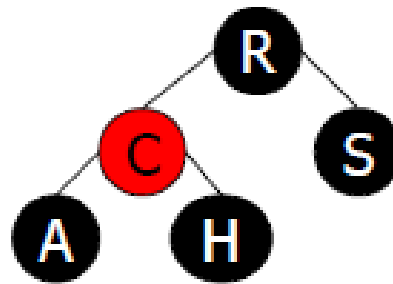
- L'inserimento richiede operazioni locali di "ristrutturazione" dell'albero con:
 - cambi di colore
 - ribilanciamento
- scendendo nell'albero si scompongono i 4-nodi, invertendo il colore dei 3 nodi risultanti
- risalendo si eseguono le rotazioni se necessario

Algoritmo di inserimento: esempio 1

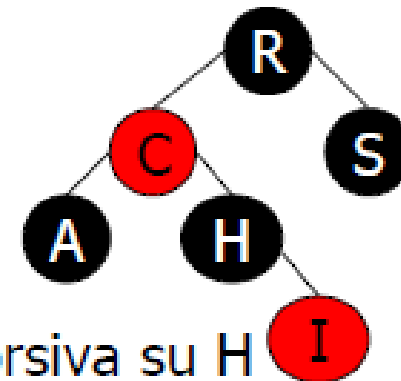
Inserimento di I



discesa ricorsiva su C
e cambio colore



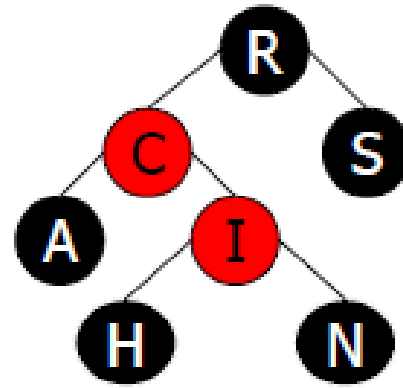
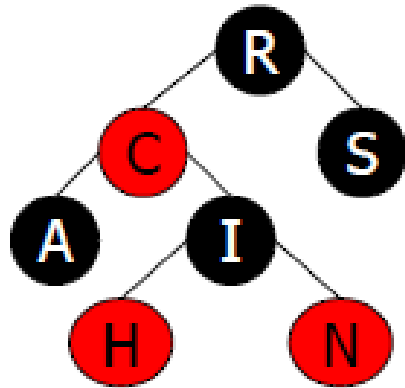
discesa ricorsiva su H
e creazione di nuovo
nodo rosso



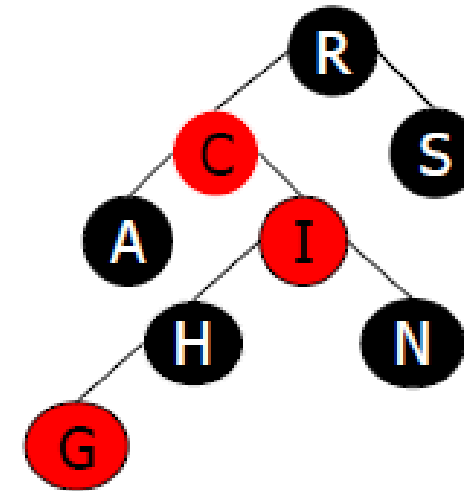
risalita senza rotazioni

Algoritmo di inserimento: esempio 2

Inserimento di G

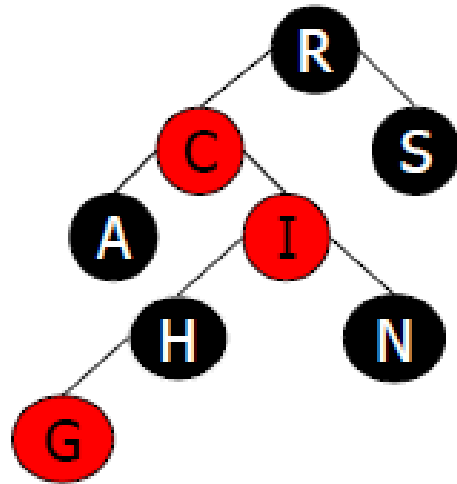


discesa ricorsiva su I
e cambio colore

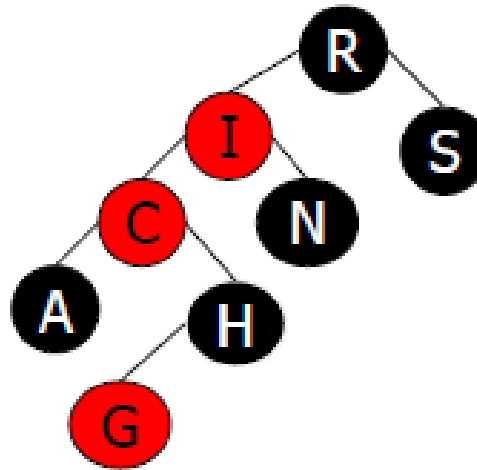


discesa ricorsiva su H
e creazione di nuovo
nodo rosso

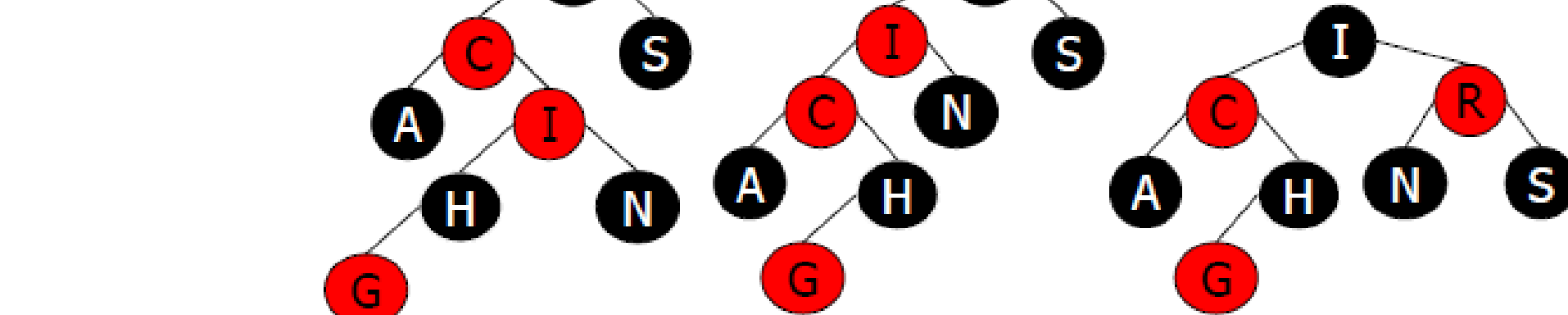
Algoritmo di inserimento: esempio 2 (cont.)



risalita e rotazione
a SX rispetto a C

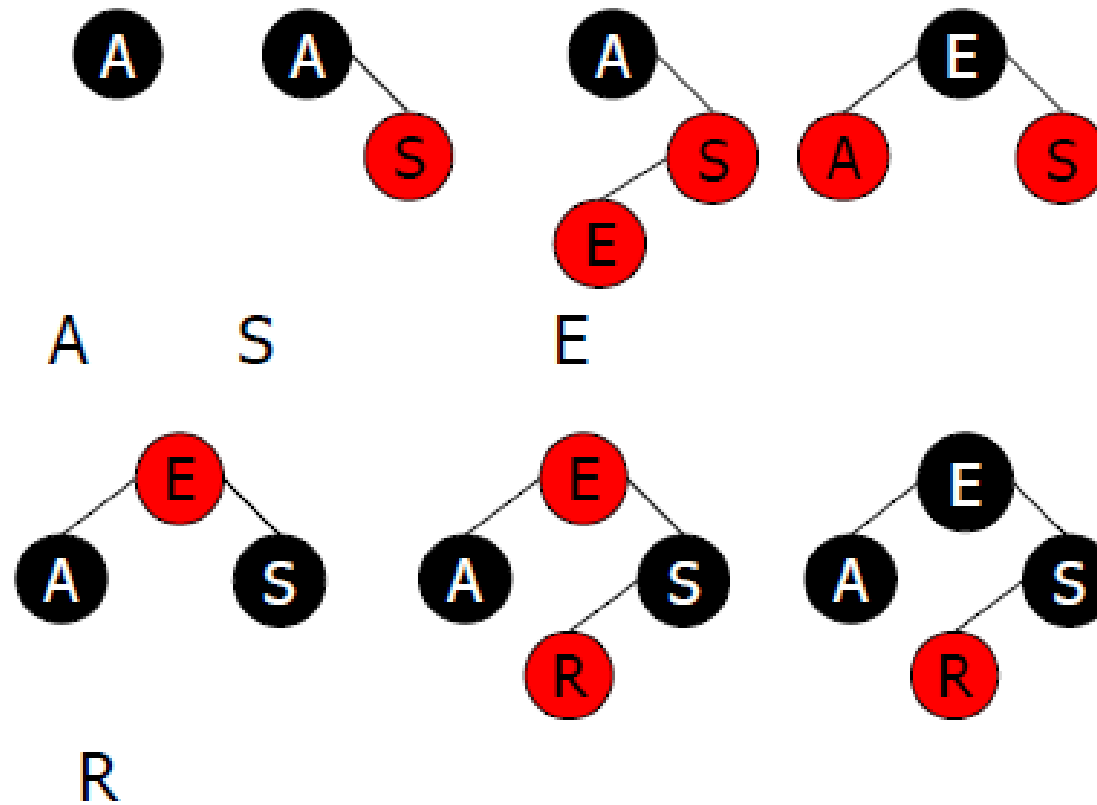


risalita e rotazione
a DX rispetto a R
cambio colore della radice

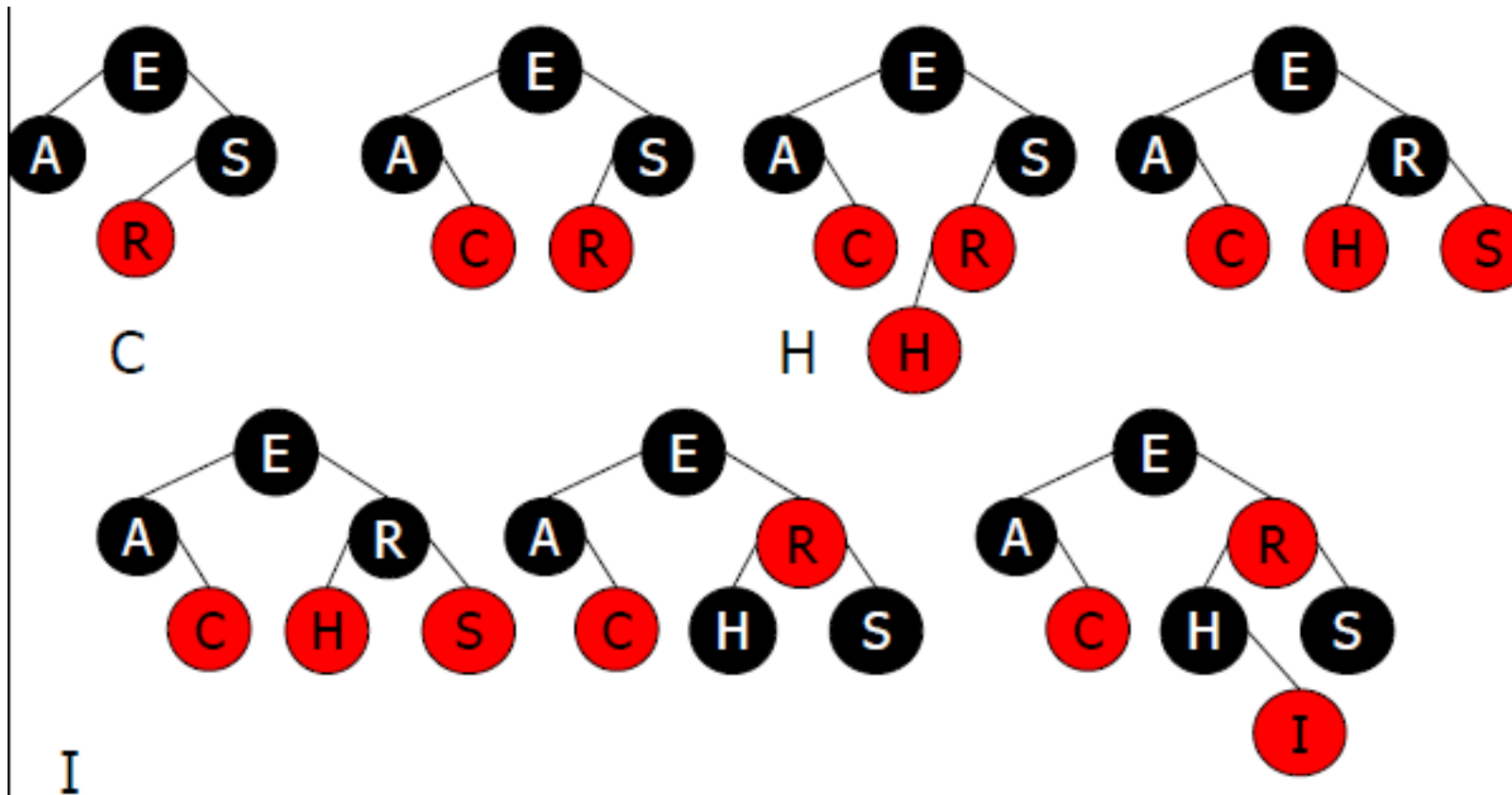


Algoritmo di inserimento: esempio 3

Inserimento di A S E R C H I in RB-tree vuoto

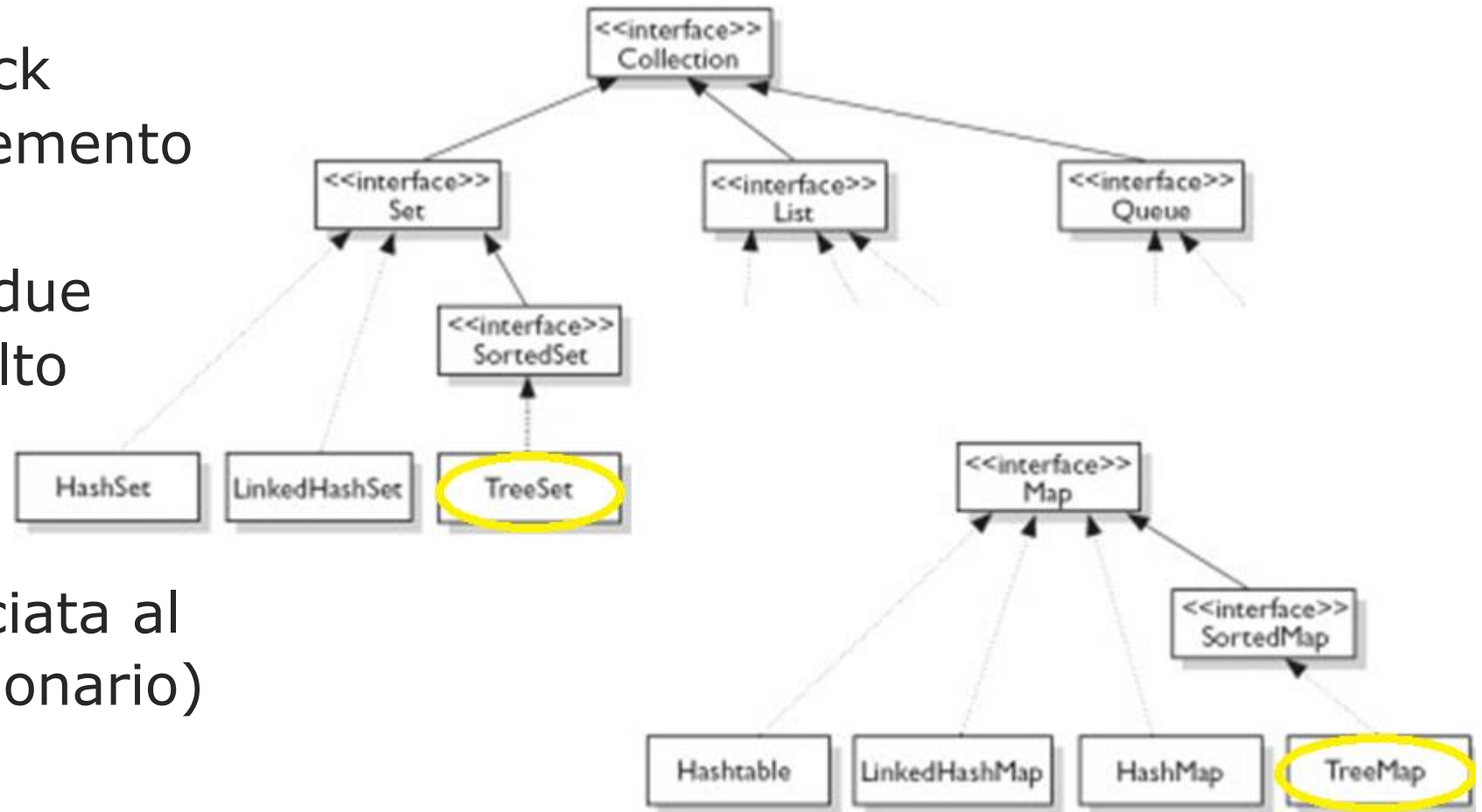


Algoritmo di inserimento: esempio 3 (cont.)



Classi TreeSet e TreeMap

- Gli alberi red-black costituiscono l'elemento centrale per la realizzazione di due classi del JCF molto usate:
 - **TreeSet**
 - **TreeMap** (associata al tipo di dato Dizionario)





UNIVERSITÀ
DEGLI STUDI
DELL'AQUILA



DISIM
Dipartimento di Ingegneria
e Scienze dell'Informazione
e Matematica



Domande?

Giovanna Melideo
Università degli Studi dell'Aquila
DISIM