

STRUTTURE DATI PER INDICI:

Struct Auxiliarie per Velocizzare Accesso:

M AD ALBERO (Vtta Scorsa)

- RIEMPIMENTO $\leftarrow \frac{\text{MAX}}{\text{MIN}}$

OPERAZIONE RICERCA SU BT TREE:

Ricerca Valore di Chiave K.

PASSO 1: Cercare nel nodo Radice il più Piccolo Valore di chiave Maggiore di K

Se $K = 15$



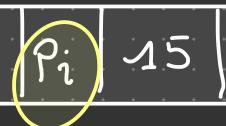
(15)
 $K_i > K$

punta al SOTTOALBERO con chiavi
 $15 \leq K_j \leq 19$

Se K_i esiste Seguo il PUNTATO P_i , altrimenti Seguo il PUNTATORE P_m che so puntare al SOTTOALBERO che ha Valori di Chiave $K_j \geq 19$

PASSO 2: Continua Come Passo 1 finché non Raggiungo il Nodo foglia.

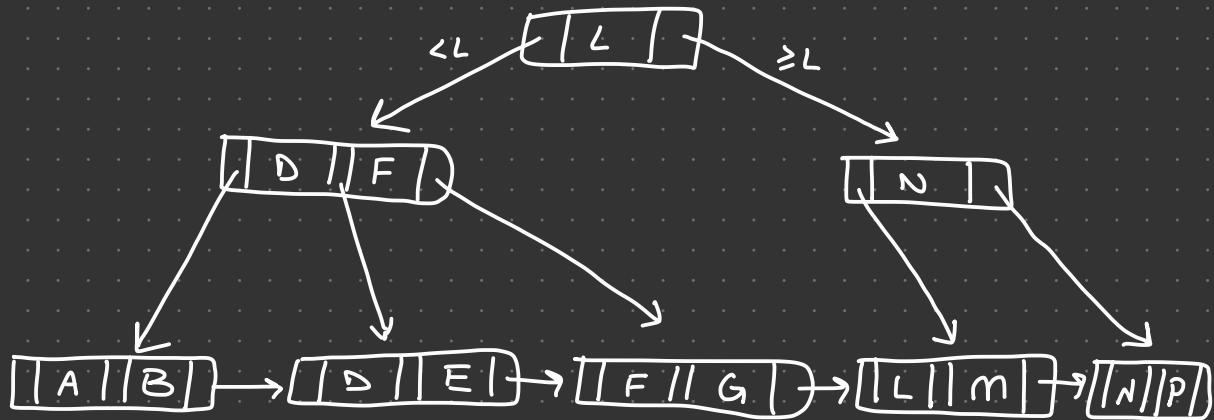
Allora cerco il Valore di chiave K e accedo alle TUPLE



PUNTA ALCE TUPLE con Valore di Chiave K=15

Partendo da Questo Albero:

Alla RICERCA DI G:



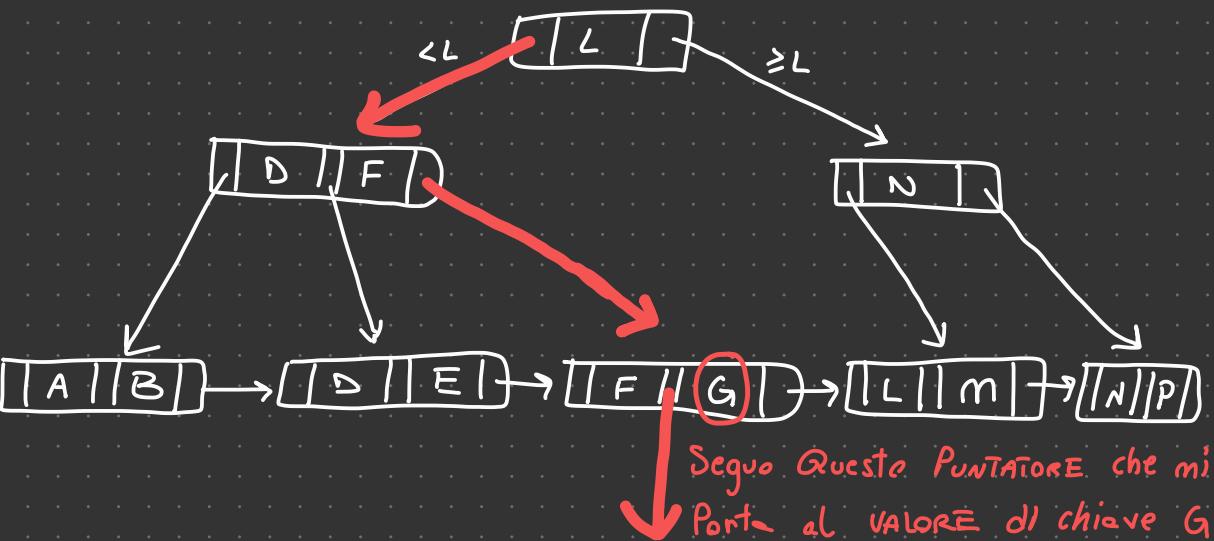
FAN-OUT = 4, rispetto ai vincoli di RIEMPIMENTO: $2 \leq \# \text{chiavi} \leq 3$

soglia

$2 \leq \# \text{puntatori} \leq 4$
intermedi

SEGUO QUESTI RAMI

Alla RICERCA DI G:



Per Verificare STRUTTURA Partice dai Nodi foglia e Salire

BILANCIAMENTO \Rightarrow Garantire Stessa profondità Su TUTTI i Nodi

Mi dà il COSTO della Ricerca fatta \leftarrow

Tramite INDICE

#Accessi Alla MEM. SECONDARIA

Complessità su Albero Bilanciato è Logaritmica sul NUMERO dei NODI.

Profondità di un B+Tree $\leq 1 + \log_2 \lceil \frac{n}{2} \rceil \left(\frac{\# \text{Valori Chiave}}{\lceil \frac{n-1}{2} \rceil} \right)$

Dove:

$\# \text{Valori Chiave} = \# \text{Chiavi da Inserire nell' Albero}$

Qual' è il #Max di nodi foglia? $N_{fmax} = \frac{\# \text{Valori Chiave}}{\lceil \frac{n-1}{2} \rceil}$

Vincolo Riempimento Minimo

Posso ora CALCOLARE I LIVELLI:

$N_{max,lvl}$ necessari per Gestire N_{fmax} foglie è:

$$N_{lvlmax} = \log \lceil \frac{n}{2} \rceil (N_{fmax})$$

INSEMENTO B+TREE:

Può MODIFICARE Struttura dell' Albero

PASSO 1: Ricerca il NODO FOGLIA N_f dove K deve ESSERE INSERITO (uso OP. DI RICERCA)

PASSO 2: Se K è Presente in N_f :

■ Se INDICE è PRIMARIO allora Nessuna Azione

■ Altrimenti si Aggiorna il BUCKET dei Puntatori

Altrimenti inserisco K rispettando l'ORDINE delle CHIAVI

■ INDICE PRIMARIO inserire puntatore Alla TUPLA con Chiave K

■ ALTRIMENTI Si Crea Nuovo BUCKET di puntatori Contenente il PUNTATORE Alla TUPLA

$n = 5$

$K = 15$

$NF =$	<table border="1"><tr><td>3</td><td> </td><td>8</td><td> </td><td>12</td><td> </td><td>16</td><td> </td><td>18</td></tr></table>	3		8		12		16		18
3		8		12		16		18		

n Viola.

$NF' =$	<table border="1"><tr><td>3</td><td> </td><td>8</td><td> </td><td>12</td><td> </td><td>15</td><td> </td><td>16</td><td> </td><td>18</td><td> </td></tr></table>	3		8		12		15		16		18	
3		8		12		15		16		18			

P_{15}

Devo PERÒ TENER Conto DEL Vincolo Di RIEMPIMENTO MASSIMO $\Rightarrow \left\lceil \frac{n-1}{2} \right\rceil \leq \# \text{Chiavi} \leq n-1$

Se l'INSERIMENTO di un nuovo Valore di Chiave ALL'INTERNO di un Nodo SOGLIA VIOLA il Vincolo di RIEMPIMENTO MAX faccio la **SPLIT**

→ DIVIDE IN 2 Nodi UN Nodo:

① I primi $\lceil \frac{n-1}{2} \rceil$ Vanno nel Primo

② I rimanenti Nel Secondo

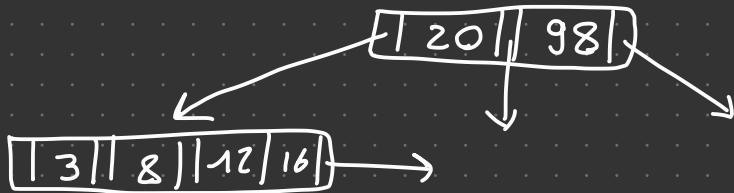
$n=5$ Quindi $\frac{5-1}{2}$ Sono nel Primo



③ Inserire Nel Nodo Padre un PUNTATORE al Secondo Nodo
Aggiustando i Valori di Chiave presenti nel PADRE

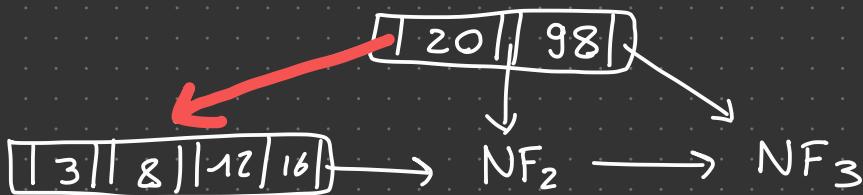
④ PROPAGARE L'AGGIORNAMENTO \Rightarrow Split (e Volte Aggiungere Anche un Interco livello)

ESEMPIO:



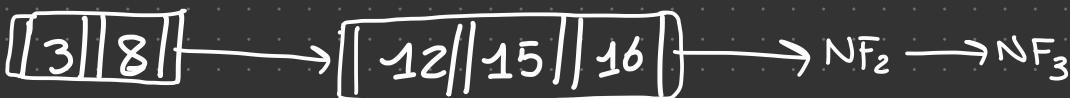
Voglio Inserire $K=15$ con fan-out $n=5$

- ③ Ricerca il Val di chiave Immediatamente dopo 15 (che è 20)

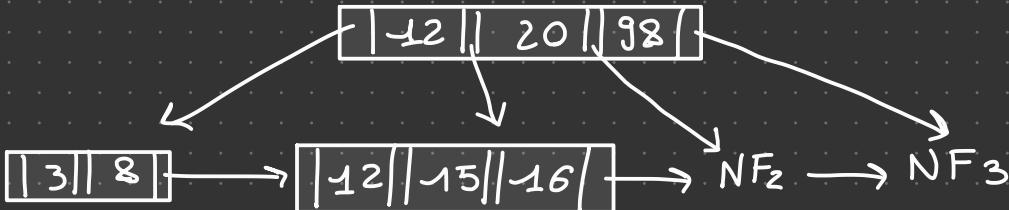


Ma $2 \leq \#Chiavi \leq 4$
NON Viene RISPETTATO

- ③ Splitto ; nodo Secondo la Regola $\lceil \frac{n-1}{2} \rceil$



- ③ Aggiorna Nodo Radice



④ Radice Rispetta i VINCOLI? Si, $2 \leq \# \text{Puntatori} \leq 5$ Quindi lo SPLIT finisce Qui, altrimenti Aurei SPLITATO la RADICE

CANCELLAZIONE BTREE:

Può Violare RIEMPIMENTO Minimo:

① Ricerca Nodo FOGLIA N_f che CONTIENE il Valore K da Cancellare

② Cancello K ed Il Suo puntatore:

■ INDICE SECONDARIO libero BUCKET dei CONTATORI

ESEMPIO:

1	3		8		12		16	
---	---	--	---	--	----	--	----	--

K=12 da Rimuovere

1	3		8		16	
---	---	--	---	--	----	--

Se Dopo la Cancellazione di K da N_f viene VIOLATO il Vincolo di RIEMPIMENTO minimo allora Si Esegue l'operazione di MERGE:

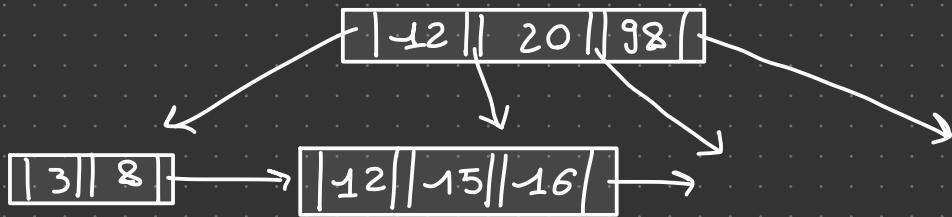
■ Va Individuato il Nodo foglia fratello (ADIACENTE) da UNIRE

■ Generare un UNICO Nodo Contenente TUTTE le Chiavi

■ Togliere un Puntatore dal PADRE ed Aggiustare gli ALTRI

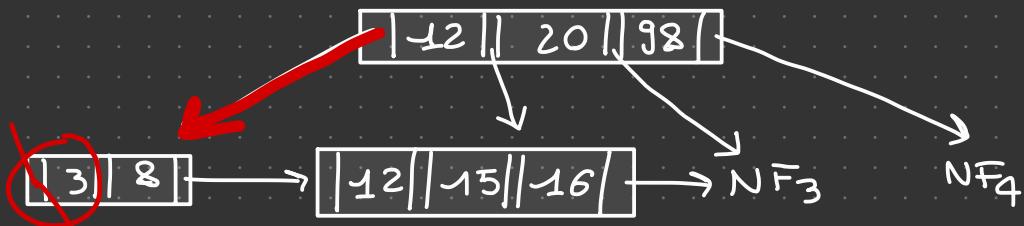
Se Anche il PADRE Viola il Vincolo Di Riempimento Minimo, Propagare il MERGE

ESEMPIO:

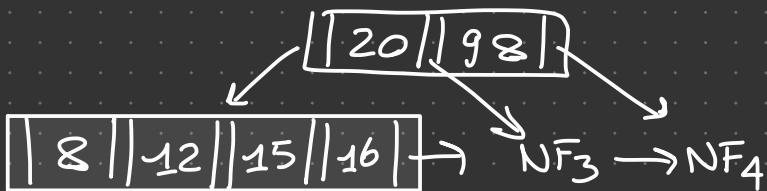


Voglio Eliminare $K=3$

④ TROVARÈ NODO



Ma VIOLA VINCOLO DI RIEMPIMENTO



DA ESAME:

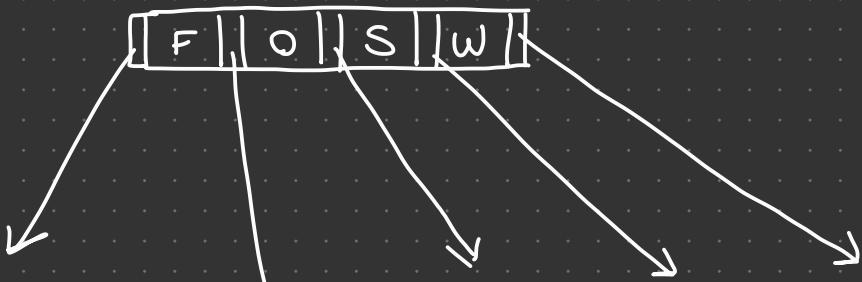
Costruire un B+Tree con fan-out = 5 con i Seguenti nodi foglia



VINCOLI:

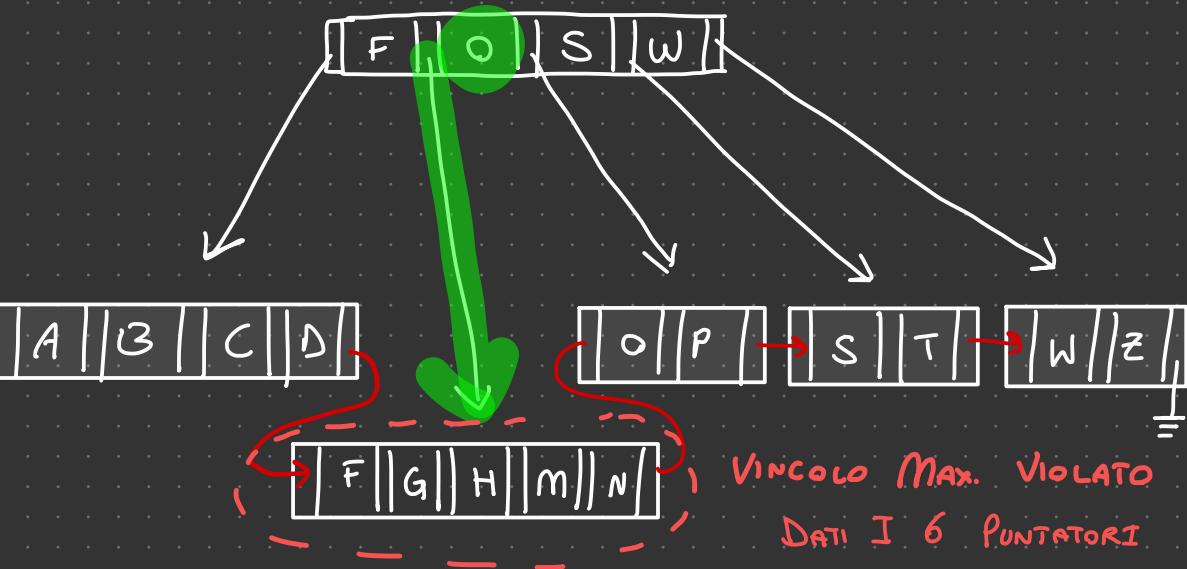
- $2 \leq \# \text{Chiavi} \leq 4$
- $3 \leq \# \text{Puntatori} \leq 5$

(A)



③ Inserire H

Ricercò il Nodo Foglia Più Piccolo Più Grande (o)



Devo fare lo SPLIT:

① 2 $[F \parallel G]$

② 3 $[H \parallel M \parallel N]$

Mi Servono 6 Puntatori Ma La RADICE non può CONTENERE

$[F \parallel H]$

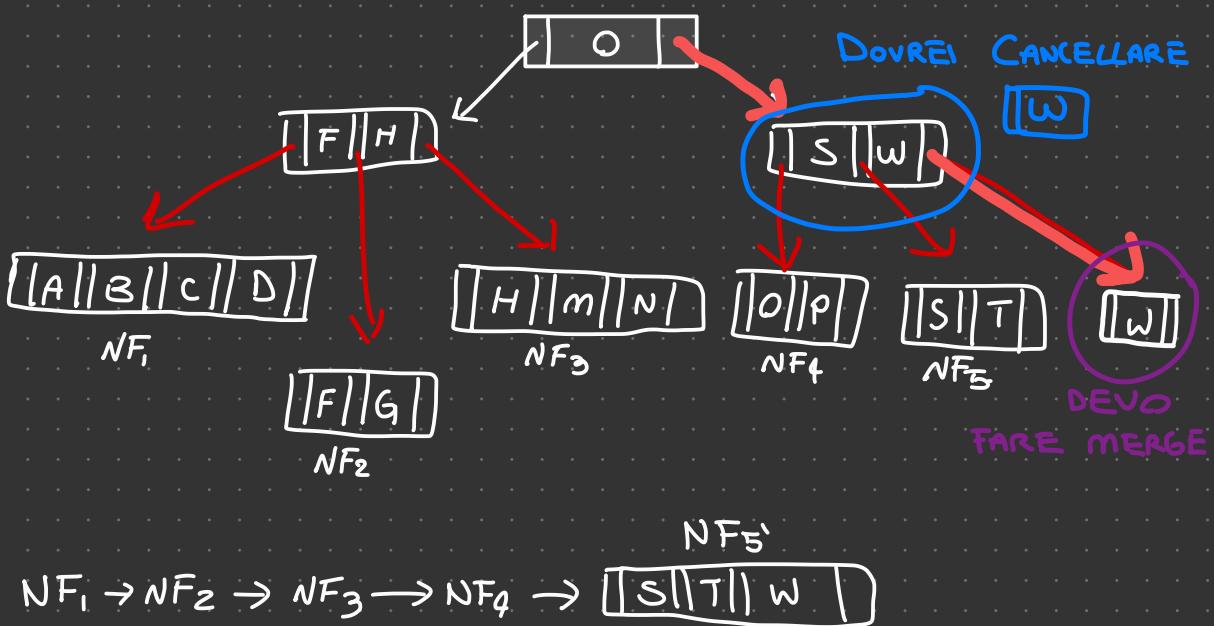
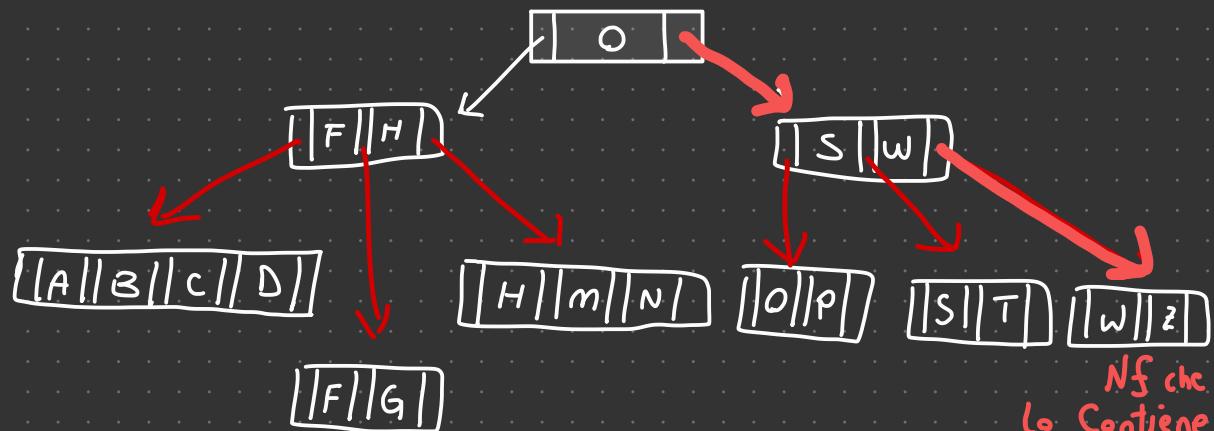
$[S \parallel W]$

Serve una Nuova Radice (Abbiamo Inserito un Nuovo livello)



devo arrivare fino in fondo all'
Albero

c) CANCELLARE il Valore Z



faendo un altro merge elimino un livello:

F || H || O || S ||