

Árvores M-Vias

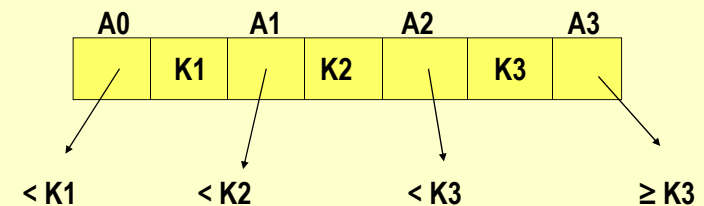
Métodos de Pesquisas de Dados

- Encontrar um dado em um conjunto de dados de forma eficiente
- Baseia-se na noção de uma **chave** (**índice**) de pesquisa
- Aplicação típica: SGBD
 - Busca de dados em disco
 - Busca de dados no *buffer* do SGBD

Árvores M-Vias

Árvore N-ária de Pesquisa (ANP)

- Cada nodo possui várias chaves



- ANP contém **m** subárvores e **n** chaves, sendo:

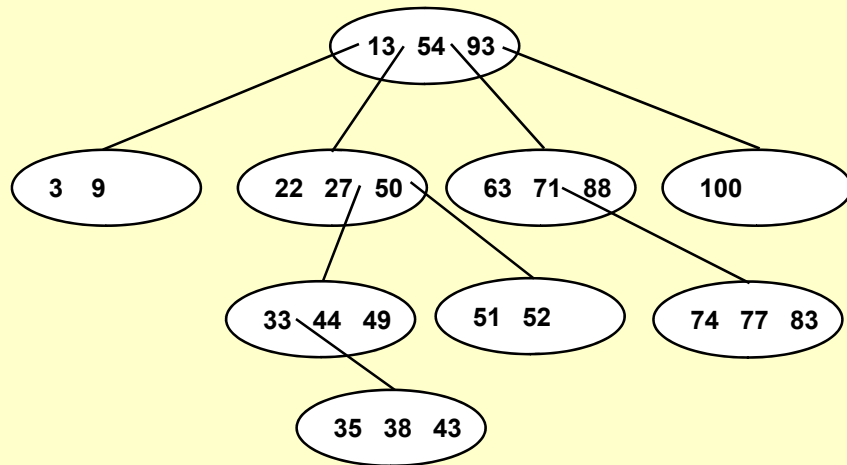
$$n = m - 1$$

$$2 \leq n \leq m$$

Árvore N-ária de Pesquisa (ANP)

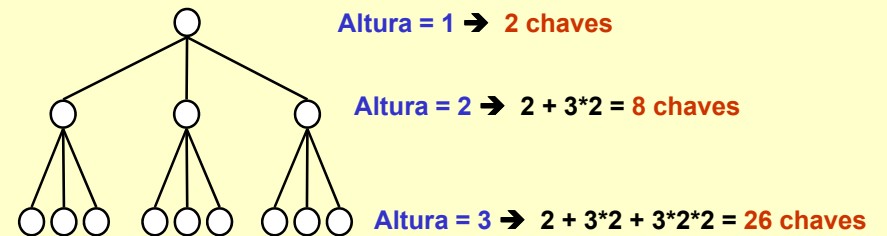
EXEMPLO

4 SUBÁRVORES
3 CHAVES POR NODO



Árvore N-ária de Pesquisa (ANP)

VANTAGENS



Quanto maior o número de subárvores, maior é o número de chaves que se pode indexar, e conseqüentemente, encontra-se uma chave com menos acesso a árvore

Árvore N-ária de Pesquisa (ANP)

VANTAGENS

- **Nodo** pode ser equivalente ao **fator de bloco do disco**
 - torna-ser mais **eficiente** o **número de acessos à disco** para a carga de dados e busca de chave em um arquivo de índices em um BD
 - exemplo:
 - fator de bloco = x bytes \approx nodos de ANP c/ 127 chaves
 - se $N = 128 \rightarrow$ cada acesso traz um nodo da ANP
 - 2o. acesso: $127^2 - 1$ chaves
 - No. acesso: $127^n - 1$ chaves

Árvore N-ária de Pesquisa (ANP)

OPERAÇÕES

- Pesquisa
 - pesquisa todos os nodos
 - pesquisa pelo valor da chave
- Incluir (chave, dado)
- Excluir (chave)

Árvore N-ária de Pesquisa (ANP) OPERAÇÕES

- Pesquisa todos os nodos
 - em **profundidade**
 - retorna ordenadamente todas as chaves

Árvore N-ária de Pesquisa (ANP) OPERAÇÕES

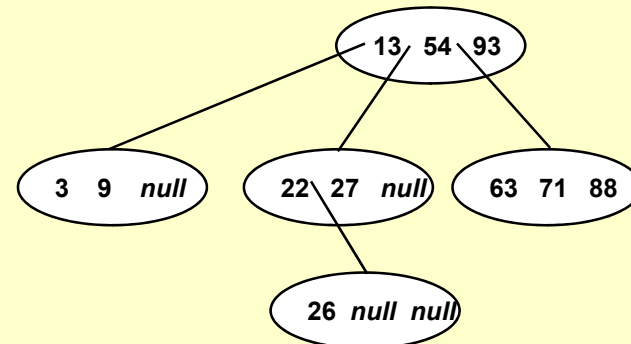
- Pesquisa pelo valor da chave
 - chave < chave pesquisada
 - pesquisa esquerda
 - chave > chave pesquisada
 - pesquisa direita
 - complexidade adicional
 - varredura das chaves em cada nodo da árvore
 - Tipos de pesquisa
 - seqüencial ou binária

Árvore N-ária de Pesquisa (ANP) OPERAÇÕES

- Inclusão
 - busca a posição na qual a chave deve ser inserida
 - nodo mais próximo com espaço para colocação da chave
 - no nodo em que a chave for inserida, o vetor deve ser rearranjado
 - deslocamento de chaves, dados e subárvores

Árvore N-ária de Pesquisa (ANP) Exercício

- Dada a árvore ANP abaixo, inserir **18** e **90**

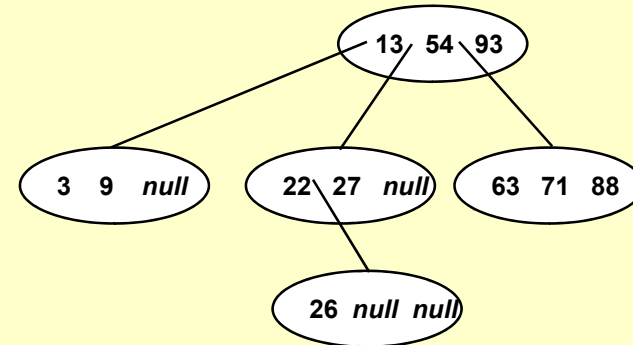


Árvore N-ária de Pesquisa (ANP) OPERAÇÕES

- Exclusão
 - se a chave não possui subárvore a ESQ e DIR vazias, ela é removida e ocorre o deslocamento no vetor para ajustar as chaves e os dados restantes
 - se a chave possui subárvore a ESQ e/ou DIR com chaves, ela é trocada com a maior chave da subárvores ESQ ou a menor chave da subárvore DIR
 - processo recursivo até que a chave não contenha subárvore

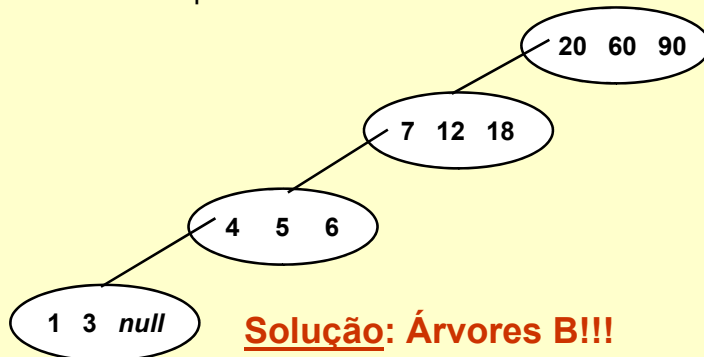
Árvore N-ária de Pesquisa (ANP) Exercício

- Dada a árvore ANP abaixo, excluir **26** e **54**



Árvore N-ária de Pesquisa (ANP) Problemas

- A árvore pode ficar desbalanceada
 - Exemplo: inserir 20-60-90-12-7-18-5-4-6-1-3

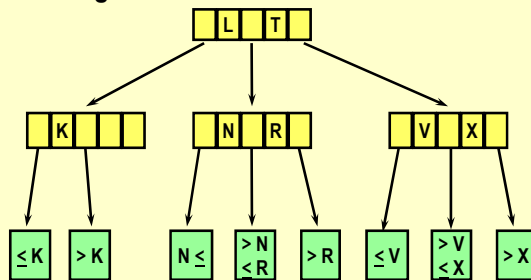


Solução: Árvores B!!!

Árvores B-Tree

B-Tree

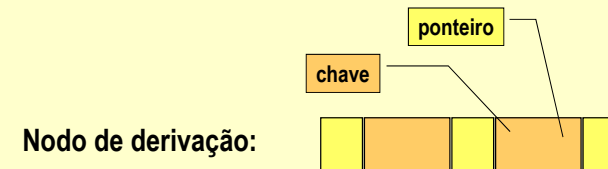
- Uma árvore B é uma árvore ANP com as seguintes características:
 - raiz com $2 \leq \text{GRAU} \leq N$
 - outros nodos tem $\lceil N/2 \rceil \leq \text{GRAU} \leq N$
 - nodos vazios estão todos na **mesma profundidade**
 - informações nas folhas
 - as folhas constituem o arquivo (ou tabela) propriamente dito, ou listas de endereços de registros



Renata de Matos Galante, Clesio S. Santos, Nina Edelweiss e Luciana P. Nedel

Estruturas de Dados - Árvores

B-trees



Folhas: ➤ 1 só valor

345

➤ diversos valores, ordenados

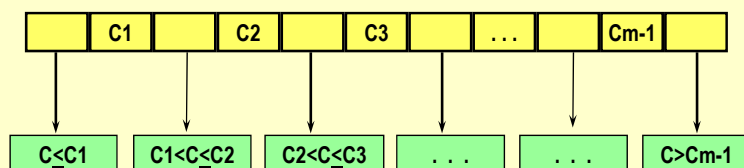
23 34 45

Renata de Matos Galante, Clesio S. Santos, Nina Edelweiss e Luciana P. Nedel

Estruturas de Dados - Árvores

B-trees

- em uma B-tree de ordem m , cada nodo possui k apontadores e $k-1$ chaves, sendo $k \leq m$

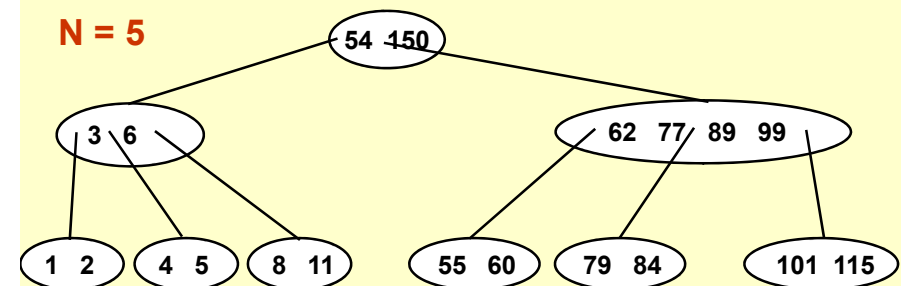


Renata de Matos Galante, Clesio S. Santos, Nina Edelweiss e Luciana P. Nedel

Estruturas de Dados - Árvores

B-Tree Exemplo

$N = 5$



Renata de Matos Galante, Clesio S. Santos, Nina Edelweiss e Luciana P. Nedel

Estruturas de Dados - Árvores

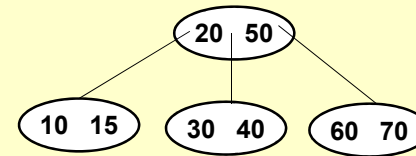
B-Tree

Inserir

- Princípio
 - inserção sempre no nodo folha
 - se há espaço no nodo, ali será inserido
- Excede capacidade
 - divisão do nodo (*split*)
 - a chave central é enviada para o pai
 - o processo se repete até que não ocorram mais divisões ou seja criada uma nova raiz
 - a árvore cresce para cima

B-Tree

Exercício



N = 5 (5 ponteiros – 4 chaves)

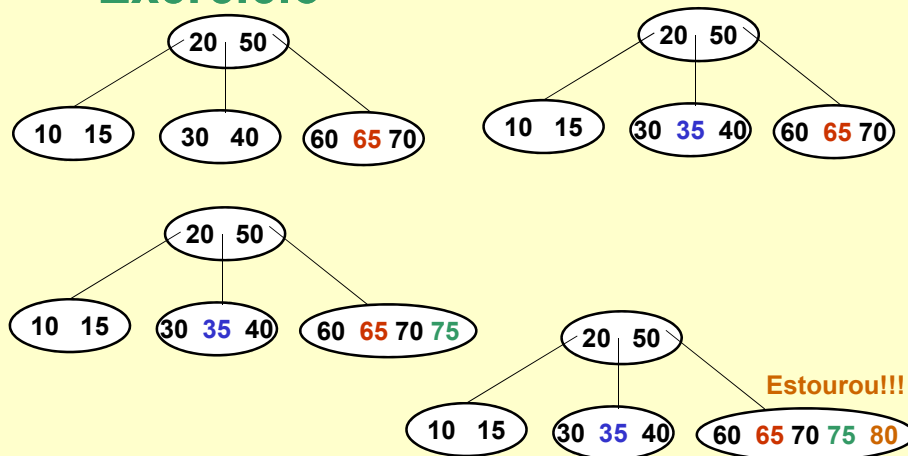
- Inserir passo a passo: 65, 35, 75, 80

B-Tree

Exercício

N = 5

- Inserir passo a passo: 65, 35, 75, 80

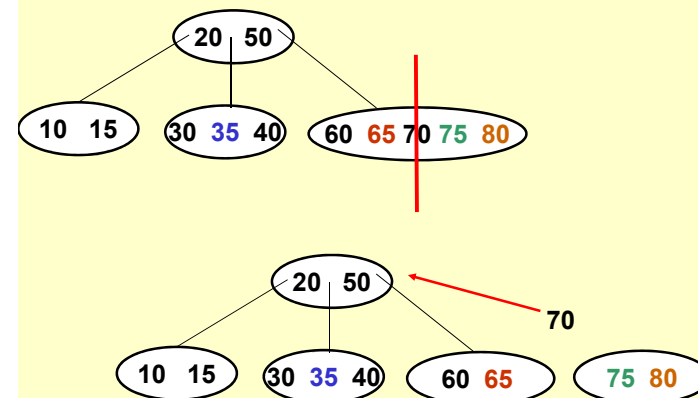


B-Tree

Exercício

N = 5

- Inserir passo a passo: 65, 35, 75, 80

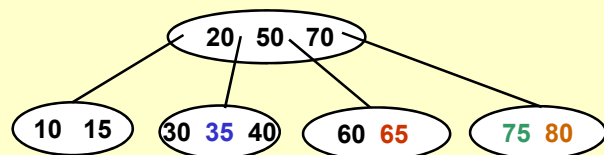


B-Tree

Exercício

N = 5

• Inserir passo a passo: 65, 35, 75, 80



Exercícios

- Dado **N = 5** (5 ponteiros – 4 chaves), mostre como fica uma **árvore B** após cada uma das seguintes inserções: 20, 10, 40, 50, 30, 55, 3, 11, 4, 28, 36, 33, 52, 17, 25, 13, 45, 9, 43, 8 e 48

B-Tree

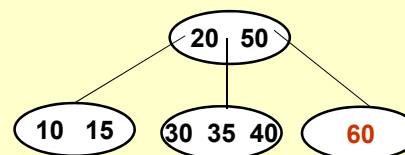
Excluir

- Princípio
 - pesquisa na árvore pela chave informada
 - se a chave não possui subárvore ESQ e DIR, então remove a chave e faz o deslocamento no vetor
 - senão troca a chave pela maior chave na subárvore ESQ ou pela menor chave na subárvore DIR e recursivamente exclui a chave na subárvore para qual ela foi enviada

B-Tree

Excluir

- Após remoção, é possível que o número de chaves em um nodo não raiz seja menor que o permitido
 - ROTAÇÃO DIREITA

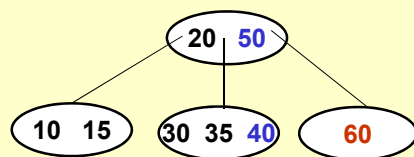


B-Tree

Excluir

- Após remoção, é possível que o número de chaves em um nodo não raiz seja menor que o permitido

– ROTAÇÃO DIREITA

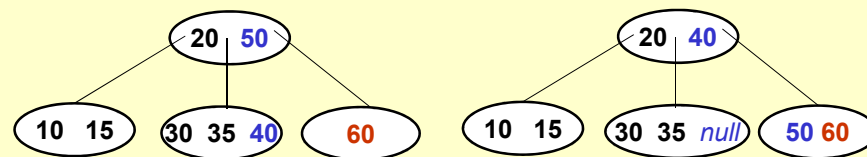


B-Tree

Excluir

- Após remoção, é possível que o número de chaves em um nodo não raiz seja menor que o permitido

– ROTAÇÃO DIREITA

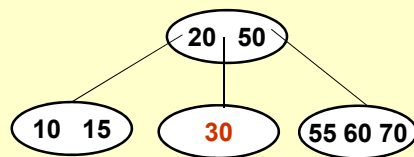


B-Tree

Excluir

- Após remoção, é possível que o número de chaves em um nodo não raiz seja menor que o permitido

– ROTAÇÃO ESQUERDA

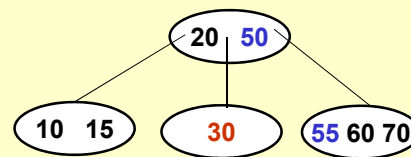


B-Tree

Excluir

- Após remoção, é possível que o número de chaves em um nodo não raiz seja menor que o permitido

– ROTAÇÃO ESQUERDA

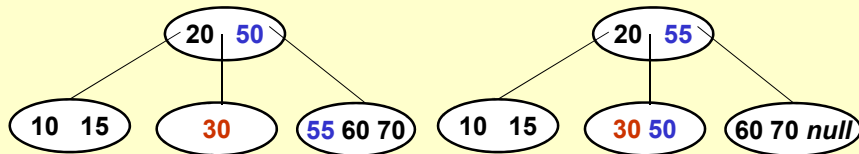


B-Tree

Excluir

- Após remoção, é possível que o número de chaves em um nodo não raiz seja menor que o permitido

– ROTAÇÃO ESQUERDA

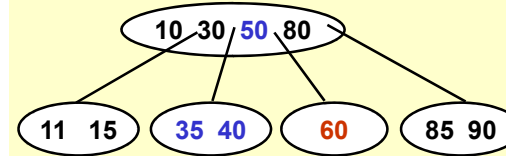


B-Tree

Excluir

- Quando não existe nodo irmão, faz a unificação (merge) de nodos.

– MERGE

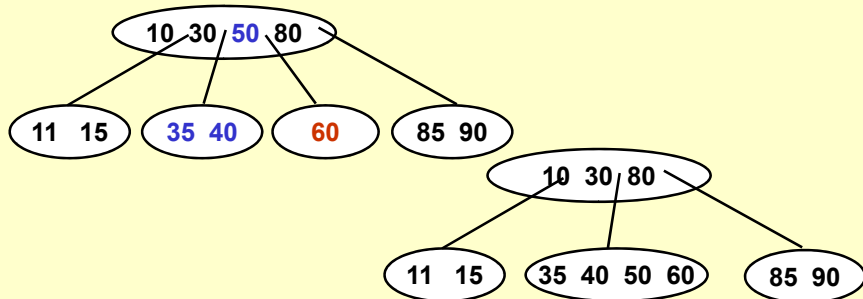


B-Tree

Excluir

- Quando não existe nodo irmão, faz a unificação (merge) de nodos.

– MERGE

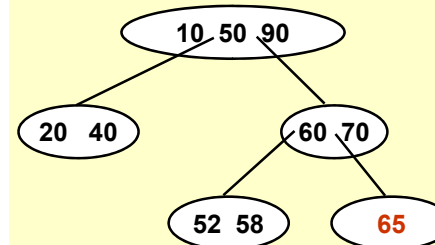


B-Tree

Excluir

- Quando não existe nodo irmão, faz a unificação (merge) de nodos.

– MERGE

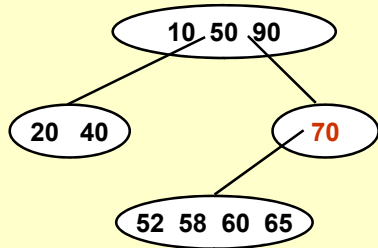


B-Tree

Excluir

- Quando não existe nodo irmão, faz a unificação (merge) de nodos.

– MERGE

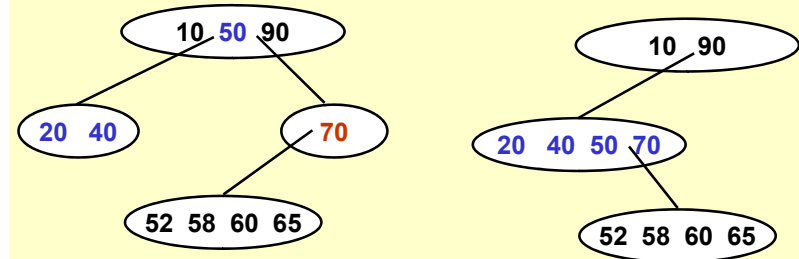


B-Tree

Excluir

- Quando não existe nodo irmão, faz a unificação (merge) de nodos.

– MERGE



Exercícios

- Agora mostre como fica a árvore após as seguintes exclusões: 20, 33, 4, 50, 30, 55, 11, 40, 28, 36, 10, 52, 17, 25, 13, 45, 9, 43, 8

Exercícios

- <http://slady.net/java/bt/view.php>