

ATIVIDADE SOBRE ÁRVORES B

1. Definição e Propriedades de Árvores B (0,25 ponto)

-Ordem da árvore: A ordem de uma árvore B, geralmente denotada como m , determina o número máximo de filhos que um nó pode ter. Uma árvore B de ordem m pode ter no máximo $m-1$ chaves por nó e, exceto a raiz, cada nó deve ter no mínimo $\lceil m/2 \rceil - 1$ chaves.

-Número de chaves em cada página (nó): Cada nó (página) de uma árvore B pode conter entre $\lceil m/2 \rceil - 1$ e $m-1$ chaves. Isso significa que, por exemplo, em uma árvore B de ordem 5, cada nó pode ter entre 2 e 4 chaves.

-Profundidade da árvore: Todas as folhas de uma árvore B estão no mesmo nível, o que garante que a profundidade seja uniforme. A profundidade da árvore depende da quantidade de chaves inseridas, mas é mantida aproximadamente logarítmica em relação ao número total de chaves.

-Balanceamento da árvore: A árvore B é mantida balanceada por meio de operações de cisão e fusão. Sempre que um nó excede o número máximo de chaves, ele é dividido em dois e a chave do meio sobe para o nó pai. Da mesma forma, se um nó ficar com menos chaves do que o permitido, pode ocorrer uma redistribuição ou fusão com nós irmãos para manter o balanceamento.

2. Inserção em uma Árvore B (0,25 ponto)

Assumindo que a árvore B trabalhada é de ordem 4, ou seja, cada nó pode ter no máximo 3 chaves. As chaves fornecidas para inserção são: 15, 5, 1, 20, 25, 30, 7, 10, 35.

-Inserção de 15: A árvore está vazia, então 15 é inserido diretamente:

[15]

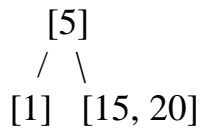
-Inserção de 5: Agora temos duas chaves, sem a necessidade de cisão:

[5, 15]

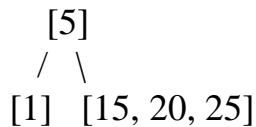
-Inserção de 1: A página pode suportar mais uma chave:

[1, 5, 15]

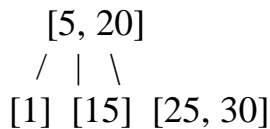
-Inserção de 20: A página agora tem 4 chaves, excedendo o limite de 3. Portanto, dividimos a página e promovemos 5:



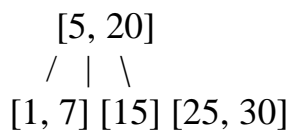
-Inserção de 25: Inserimos 25 na página da direita:



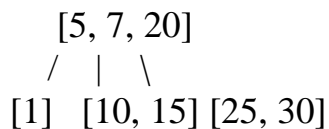
-Inserção de 30: A página da direita agora tem 4 chaves. Dividimos a página e promovemos 20:



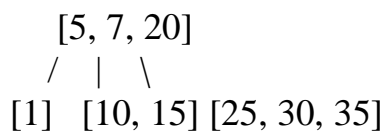
-Inserção de 7: Inserimos 7 na página da esquerda:



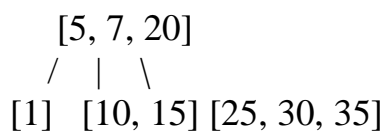
-Inserção de 10: A página da esquerda agora tem 3 chaves. Dividimos a página e promovemos 7:



-Inserção de 35: Inserimos 35 na página da direita:



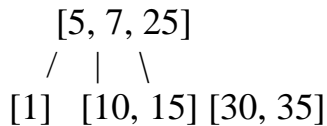
Estado final da árvore:



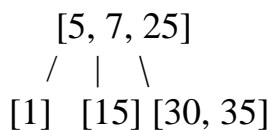
3. Remoção em uma Árvore B (0,25 ponto)

Agora, o início da remoção das chaves 20, 10 e 1, mantendo o balanceamento:

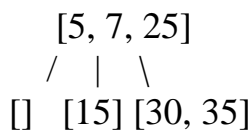
-Remoção de 20: A chave 20 é removida, e 25 é promovido para substituir 20:



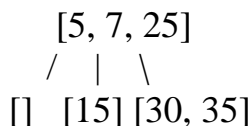
-Remoção de 10: A chave 10 é removida da página central:



-Remoção de 1: A chave 1 é removida. Como restam chaves suficientes, a estrutura permanece balanceada:



Estado final após remoções:



4. Busca em uma Árvore B (0,25 ponto)

Realizando a busca pelas chaves 25 e 8.

-Busca de 25: Começando pela raiz, comparando 25 com as chaves na raiz. Como 25 é igual a uma das chaves na raiz, a busca é bem-sucedida.

Busca de 8: Começando pela raiz, e 8 é menor que 25, então segue-se para a página do meio. Lá, compara-se 8 com as chaves 7 e 15. Como 8 não existe nessa página e não há mais subárvores para descer, a busca é infrutífera, e conclui-se que a chave 8 não está na árvore.