



Árvores B e B+

Gabriela Vitória Aquino Pereira

Clebson Pereira dos Santos

Wallan de Melo Lima

O que são árvores B e B+?

- Árvores B e B+ são soluções ideais para cenários onde há grande volume de dados e a necessidade de realizar operações de busca e atualização de forma eficiente.
- São usadas principalmente em sistemas de banco de dados e sistemas de arquivos por causa da capacidade de armazenar dados de maneira eficiente e manter-se balanceadas.

Árvores B

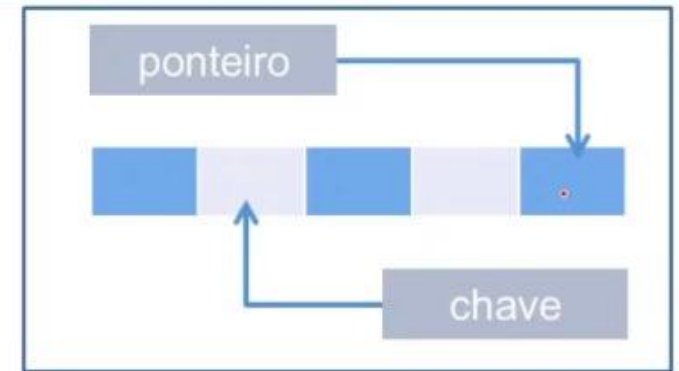
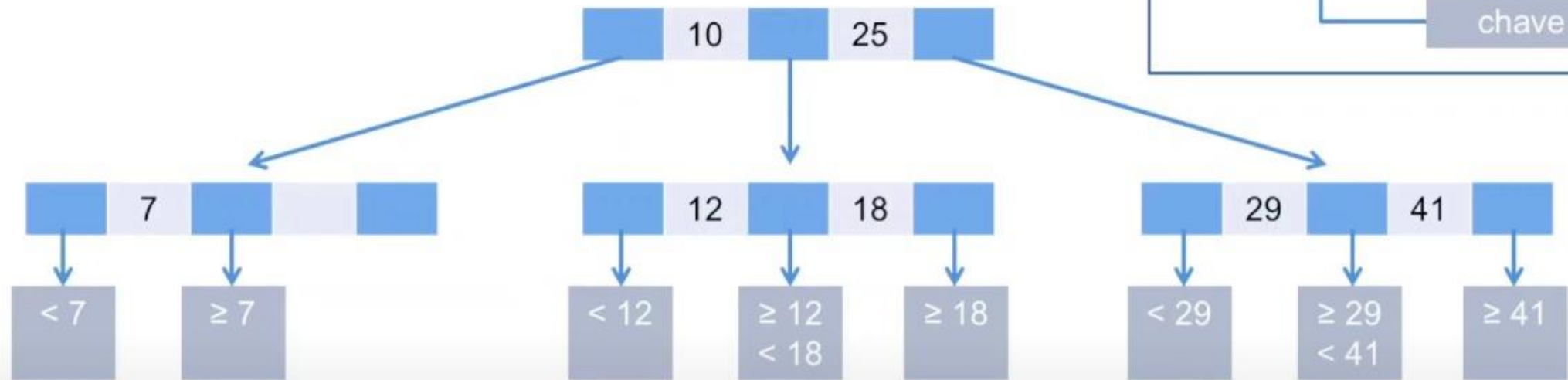
Definição e estrutura

- **Árvore B** é uma árvore de busca balanceada que mantém seus dados ordenados e permite buscas, inserções, deleções e varreduras de maneira eficiente.
- **Nós** da árvore B contêm várias chaves e ponteiros, diferentemente de árvores binárias, que possuem no máximo duas chaves (à esquerda e à direita).
- **Ordem da árvore B** (denotada como m) define o número máximo de filhos que um nó pode ter.
 - Cada nó interno pode ter até m filhos e armazenar até $m-1$ chaves.
 - O número mínimo de chaves em um nó é $\lceil m/2 \rceil - 1$.

Estrutura

- **Árvore B** é uma árvore de busca balanceada de ordem m onde cada nó pode ter até m filhos.
- Uma árvore B de ordem m segue as seguintes regras:
- Cada nó interno pode conter até $m-1$ chaves.
- O número mínimo de filhos de um nó interno é $\lceil m/2 \rceil$.
- Todos os nós folha estão no mesmo nível.
- O nó raiz deve ter no mínimo dois filhos (se não for uma folha).
- A árvore mantém-se balanceada com operações de **split** (divisão) e **merge** (fusão).

Árvores de múltiplos caminhos



Altura da árvore

A altura da árvore B, h , é logarítmica em relação ao número total de elementos n , com base na ordem m :

$$h \leq \log_{\lceil m/2 \rceil} \frac{n+1}{2}$$

Isso garante que as operações de busca, inserção e remoção possam ser realizadas em tempo $O(\log n)$.

Propriedades de uma árvore B

- Todos os nós folha estão no mesmo nível (árvore balanceada).
- Um nó com n filhos tem $n-1$ chaves.
- As operações mantêm a árvore balanceada, o que garante que a altura da árvore seja logarítmica, garantindo eficiência.

Operações

Busca:

Começa na raiz e segue pelos nós até encontrar a chave ou um nó folha onde ela deveria estar.

Comparações são feitas para decidir qual caminho seguir nos nós internos.

Inserção:

A chave é inserida na posição correta no nó folha.

Se o nó exceder o número máximo de chaves permitido, ocorre um **split** (divisão do nó), promovendo uma chave para o nível superior.

Remoção:

Quando uma chave é removida, a árvore pode precisar ser reorganizada para manter as propriedades da árvore B.

Caso a remoção esvazie um nó além do mínimo permitido, ocorre um **merge** (fusão) com um nó adjacente ou redistribuição das chaves.

Complexidade de tempo

Complexidade de Tempo em Notação big O

Algoritmo	Caso Médio	Pior Caso
Espaço	$O(n)$	$O(n)$
Busca	$O(\log n)$	$O(\log n)$
Inserção	$O(\log n)$	$O(\log n)$
Remoção	$O(\log n)$	$O(\log n)$

Árvores B+

Definição e estrutura

- A **árvore B+** é uma variante da árvore B com algumas modificações que a tornam ainda mais eficiente para sistemas de banco de dados.
- **Diferenças em relação à árvore B:**

Todos os valores reais (dados) são armazenados nas folhas.

Os nós internos contêm apenas ponteiros e chaves que servem de índice, facilitando a busca.
- As folhas são conectadas entre si por meio de ponteiros, formando uma **lista encadeada**. Isso permite percorrer os dados de maneira sequencial mais eficiente.

Estrutura do nós

- Nos nós internos, as chaves são usadas apenas para decidir o caminho de busca, e não para armazenar dados.
- Nas folhas, cada nó armazena as chaves e seus respectivos ponteiros para os dados. As folhas estão organizadas de forma que cada folha tenha um ponteiro para a próxima, formando uma **lista encadeada**.

Propriedades adicionais:

- Os nós internos contêm m filhos e $m-1$ chaves.
- Os nós folha armazenam entre $\lceil m/2 \rceil$ e $m-1$ chaves.
- As folhas estão todas no mesmo nível, como na árvore B.
- **Lista encadeada** entre as folhas facilita buscas sequenciais e varreduras.

Propriedades de uma árvore B+

- Os nós folha contêm todas as chaves e os ponteiros para os dados.
- Os nós internos só contêm chaves e ponteiros para outros nós, tornando-os mais leves.
- A busca é mais eficiente para leituras sequenciais, pois os nós folha são conectados diretamente, facilitando a navegação.

Operações

Busca:

Funciona de forma similar à árvore B, mas a chave é sempre encontrada em um nó folha, o que simplifica a operação.

A lista encadeada entre folhas facilita buscas em intervalos de chaves.

Inserção:

Similar à árvore B, porém as inserções são sempre feitas nas folhas. Se uma folha exceder o limite de chaves, ela é dividida e uma nova chave é promovida ao nível superior.

Remoção:

Chaves são removidas das folhas. Se a remoção de uma chave esvaziar o nó além do limite mínimo, ocorre redistribuição ou fusão, tal como na árvore B.

Complexidade de tempo

Pior caso: $\Theta(\log n)$

Caso médio: $\Theta(\log n)$

Melhor caso: $\Theta(\log n)$

Comparação entre árvore B e B+

Estrutura:

- **Árvore B:** Tanto as chaves quanto os dados podem estar nos nós internos e nas folhas.
- **Árvore B+:** Apenas as folhas contêm os dados, os nós internos são usados apenas para roteamento (chaves e ponteiros).

Desempenho:

- **Árvore B:** Levemente mais eficiente para operações de busca em chaves isoladas.
- **Árvore B+:** Melhor para varredura sequencial de dados e consulta por intervalos, muito usada em bancos de dados e sistemas de arquivos.

Uso de memória:

- **Árvore B:** Pode utilizar mais memória, já que os dados são armazenados tanto em nós internos quanto em folhas.
- **Árvore B+:** Usa menos memória nos nós internos, o que torna a árvore mais leve.

Comparação

Critério	Árvores B	Árvores B+
Armazenamento de dados	Nós internos e folhas podem conter dados	Apenas nós folha contêm dados
Busca	O caminho de busca pode terminar em um nó interno	O caminho de busca sempre termina em uma folha
Leituras sequenciais	Menos eficiente em leituras sequenciais	Otimizado com encadeamento de folhas
Espaço	Nós internos mais pesados, já que armazenam dados e chaves	Nós internos são mais leves, armazenando apenas chaves de roteamento
Aplicação	Adequada para operações de busca pontual	Ideal para bancos de dados e sistemas que requerem buscas sequenciais rápidas

Aplicações de árvores B e B+

Árvores B:

- Utilizadas em **sistemas de gerenciamento de arquivos**, onde é necessário balancear eficientemente o acesso a dados com inserções e remoções frequentes.
- Exemplos: Sistemas de gerenciamento de memória em sistemas operacionais, sistemas de cache de discos.

Árvores B+:

- Predominantemente usadas em **bancos de dados relacionais** (como MySQL e PostgreSQL) e **sistemas de arquivos modernos** (como NTFS e ext4).
- Aplicações que precisam de leituras sequenciais rápidas e eficientes, como em bancos de dados e sistemas de armazenamento de grandes volumes de dados.



Código em C