

Estrutura de Dados

Hamilton José Brumatto

Bacharelado em Ciências da Computação - UESC

7 de fevereiro de 2024

Árvores Múltiplas

Árvores M-Múltiplas de Busca

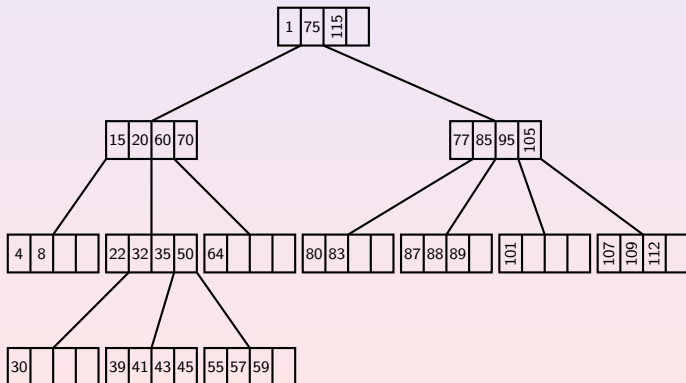
- Árvores binárias de busca são árvores classificadas em ordem
- Uma busca em uma árvore binária balanceada é $O(\log n)$.
- Mas às vezes é necessário considerar também o custo físico da busca.
- Quando trabalhamos, por exemplo, com base de dados, os dados de busca estão em disco, memória secundária.
- Ler elemento por elemento em memória secundária tem um custo alto. É mais importante uma busca em bloco.
- Precisamos de um novo modelo de árvore onde a informação não é um elemento, e sim um bloco: São as Árvores Múltiplas.

Árvores M-Múltiplas

- **Árvore de busca múltipla de ordem m**
 - Cada nó tem até m filhos e $m - 1$ chaves
 - As chaves em cada um dos nós estão em ordem não decrescente.
 - As chaves dos primeiros i filhos (e seus descendentes) não são maiores que a i -ésima chave.
 - As chaves dos últimos $m - i$ filhos (e seus descendentes) não são menores que a i -ésima chave.

Exemplo de uma árvore M-Múltipla

Árvore de busca múltipla de ordem 5:



Implementando Árvores Múltiplas

Estrutura

```
typedef struct arvore {  
    int n_chaves;  
    obj_t info[M-1];  
    struct arvore *filhos[M];  
} arvore;
```

Ações

- InserirValor:
 - Se o valor for menor que algum elemento da chave, inserir no filho anterior a esta chave. (recursivamente)
 - Se for maior que os elementos da chave, inserir na própria chave se houver espaço, ou no último filho.
- RemoverValor:
 - É preciso localizar primeiro. Sendo um elemento da chave, é preciso ver se existe o filho no seu lado direito ou no seu lado esquerdo.
 - Se não houver filhos basta deslocar próxima chave e filhos.
 - Se houver, pegar o sucessor, ou antecessor e colocar em seu lugar. (Recursivamente)

Árvores Múltiplas Balanceadas: Árvores B

- Uma Árvore B é uma árvore de busca múltipla com as seguintes propriedades:
 - A raiz tem pelo menos duas subárvores, a menos que ela seja uma folha;
 - Cada nó não-raiz e não-folha contém $k - 1$ chaves e k filhos onde $\lceil m/2 \rceil \leq k \leq m$
 - Cada nó folha contém $k - 1$ chaves, onde $\lceil m/2 \rceil \leq k \leq m$
 - Todas as folhas estão no mesmo nível

Árvores B

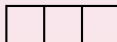
- Como falamos em memória secundária, o número de chaves costuma ser muito grande.
- Como esta árvore privilegia a leitura em bloco, a ordem também é grande.
- m tende a variar, normalmente, entre 50 e 500.
- Neste caso, mesmo para um número relativamente grande de chaves, a altura da árvore acaba sendo pequena.
- Por exemplo, para $m = 200$ e $n = 2.000.000$, temos $h \leq 4$.

Inserção de valores na árvore B

- O processo de inserção em uma árvore B é o que garante seu balanceamento.
- São as seguintes etapas:
 - A inserção é sempre realizada em uma folha, em ordem de classificação na folha.
 - Se a folha estiver cheia, cria-se uma folha irmã, a primeira fica com $\lceil m/2 \rceil$ elementos e a segunda com $\lfloor m/2 \rfloor$ elementos.
 - O último elemento da primeira folha é transferido para a (ou cria uma nova) raiz.
 - Cada uma destas folhas passam a ser filhos esquerdo e direito deste novo elemento na raiz.
 - Se ao transferir o elemento a raiz também está cheia, o processo se repete recursivamente.
 - Tomando-se o cuidado de que um filho, na escolha entre ser filho direito da última chave de um raiz ou filho esquerdo da primeira chave do outro raiz, a última escolha prevalece.

Exemplo de inserção de elementos na árvore B: $m = 4$

5 15 13 21 8 7 9 19 11 6 3 25 12



Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12

5		
---	--	--

Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12

5	15	
---	----	--

Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12

5	13	15
---	----	----

Exemplo de inserção de elementos na árvore B

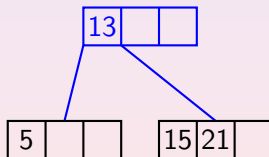
5 15 13 21 8 7 9 19 11 6 3 25 12

5	13	
---	----	--

15	21	
----	----	--

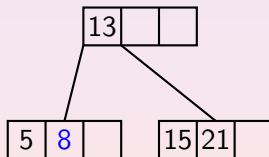
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



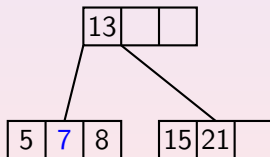
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



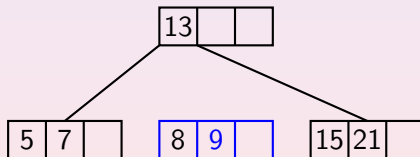
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



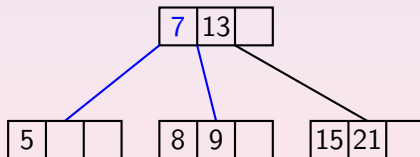
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



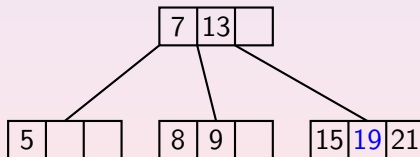
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



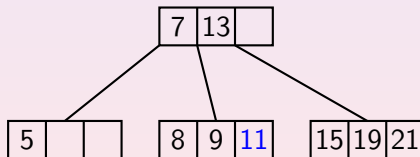
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



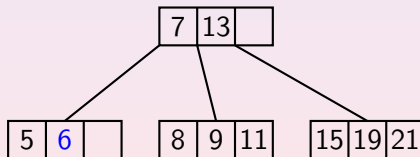
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



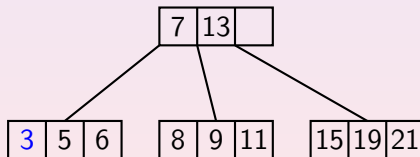
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



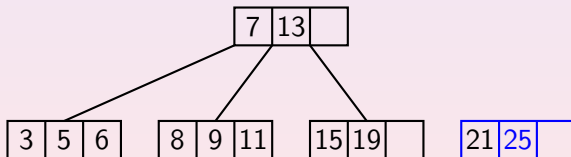
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



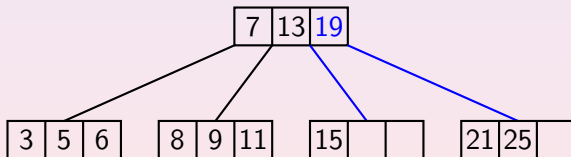
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



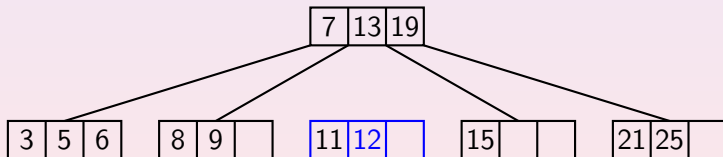
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



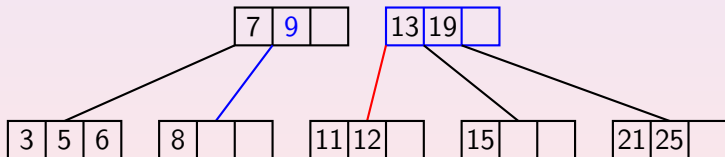
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



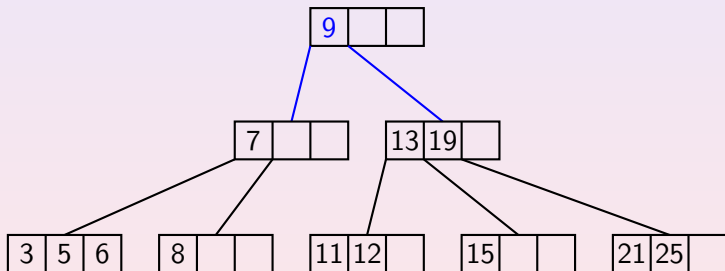
Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12



Exemplo de inserção de elementos na árvore B

5 15 13 21 8 7 9 19 11 6 3 25 12

