Estrutura de Dados (CCA410)

Aula 06 – Árvores Balanceadas (AVL)

Prof. Luciano Rossi
Prof. Leonardo Anjoletto Ferreira
Prof. Flavio Tonidandel
Prof. Fabio Suim

Ciência da Computação
Centro Universitário FEI

2° Semestre de 2025

Árvores Binárias de Busca

Propriedades

Vimos que as Árvores Binárias de Busca possuem duas propriedades fundamentais:

- Propriedade estrutural; e
- Balanceamento.

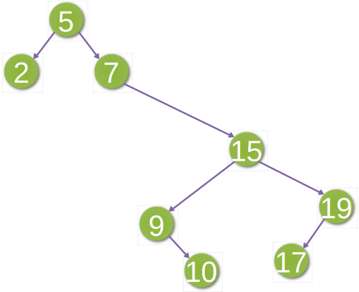
Balanceamento

Árvores Binárias de Busca

Balanceamento - Por que Árvores Balanceadas?

- A **propriedade estrutural** é observada na árvore abaixo;
- Quantas comparações são necessárias para recuperar os dados satélites do vértice de chave 7?

1	5
2	7
3	2
4	15
5	9
6	19
7	17
8	10

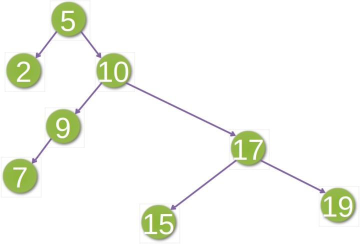


Árvores Binárias de Busca

Balanceamento – Por que Árvores Balanceadas?

- A **propriedade estrutural** é observada na árvore abaixo;
- Quantas comparações são necessárias para recuperar os dados satélites do vértice de chave 7?

1	5
2	10
3	17
4	15
5	19
6	9
7	2
8	7

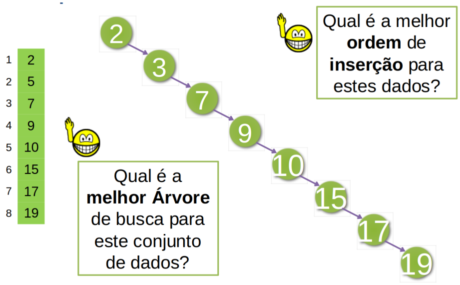


- Qual a conclusão a partir dos exemplos anteriores?

Árvores Binárias de Busca

Balanceamento – Por que Árvores Balanceadas?

- A ordem de inserção dos valores na árvore binária de busca **altera** sua organização;
- Desse modo, a eficiência de busca será impactada.



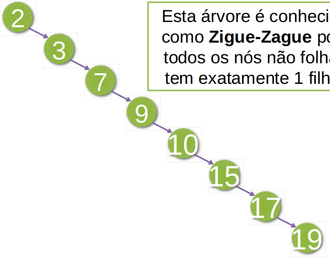
centro universitário FEI

Árvores Binárias de Busca

Balanceamento - Por que Árvores Balanceadas?

- Em uma árvore binária de busca com essa organização a complexidade de busca é da ordem de $O(n)$ (complexidade observada em listas).

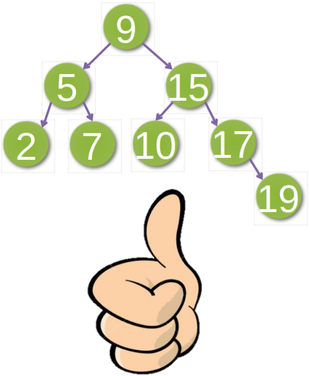
1	2
2	5
3	7
4	9
5	10
6	15
7	17
8	19

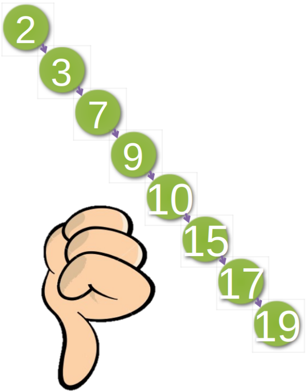


Esta árvore é conhecida como **Zigue-Zague** pois todos os nós não folhas tem exatamente 1 filho.

Árvores Binárias de Busca

Balanceamento – Qual é melhor para a busca?





Árvores AVL

Adelson-Velskii and Landis

- São árvores auto-balanceadas
- As árvores AVL garantem que:

Árvores AVL

Dado qualquer nó da árvore, a diferença entre a altura do seu ramo direito e a altura do seu ramo esquerdo é de, no máximo, uma unidade.

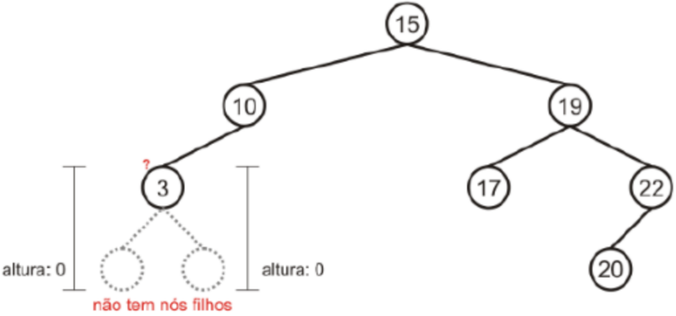
centro universitário FEI

Altura da árvore

(ou do ramo)

Árvores Binárias de Busca

Balanceamento - Árvores Balanceadas AVL Adelson-Velskii and Landis' Tree

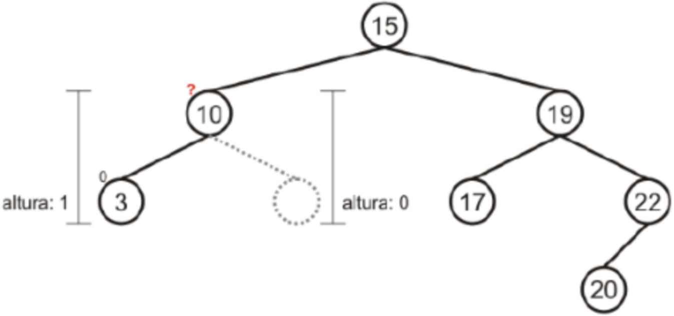


- O vértice 3 está balanceado?

centro universitário FEI

Árvores Binárias de Busca

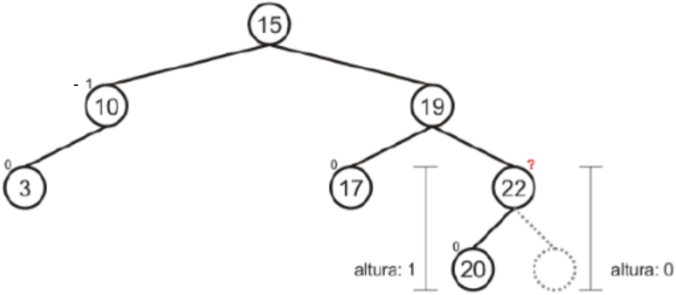
Balanceamento – Árvores Balanceadas AVL Adelson-Velskii and Landis' Tree



- O vértice 10 está balanceado?

Árvores Binárias de Busca

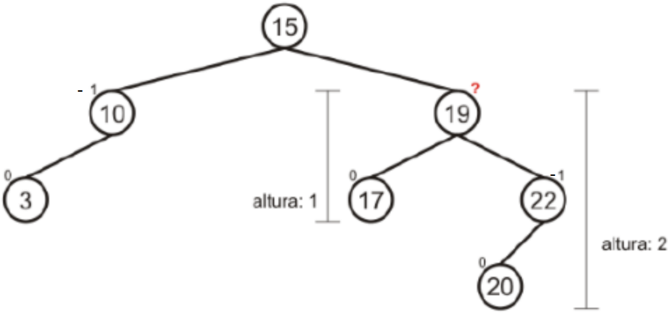
Balanceamento – Árvores Balanceadas AVL Adelson-Velskii and Landis' Tree



- O vértice 22 está balanceado?

Árvores Binárias de Busca

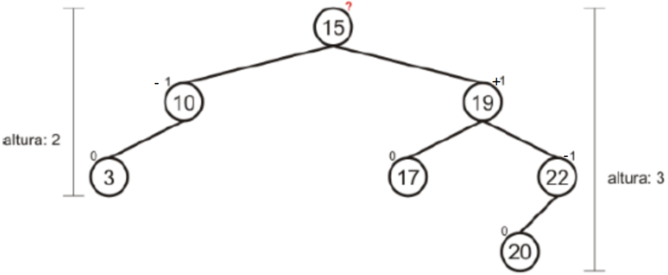
Balanceamento - Árvores Balanceadas AVL Adelson-Velskii and Landis' Tree



- O vértice 19 está balanceado?

Árvores Binárias de Busca

Balanceamento – Árvores Balanceadas AVL Adelson-Velskii and Landis' Tree



- O vértice 15 está balanceado?

Fator de Balanceamento

- O fator de balanço (fb) de um vértice v é o dado por

$$fb(v) = h_d(v) - h_e(v)$$

- Um vértice é balanceado se

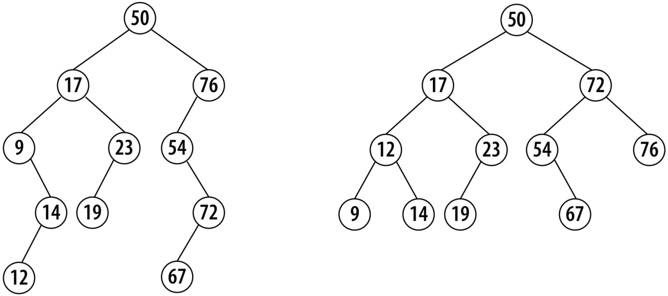
$$-1 \leqslant fb(v) \leqslant 1$$

- Uma árvore é AVL se todos os seus vértices são balanceados.

Árvores Binárias de Busca

Balanceamento - Árvores Balanceadas AVL Adelson-Velskii and Landis' Tree

- Exemplo: calcular os fatores de balanceamento das árvores abaixo:



(a) (b)

Análise pelo fator de balanceamento:

- $fb(v) = 1$: a subárvore direita é mais alta que a esquerda;
- $fb(v) = 0$: as subárvores têm alturas iguais;
- $fb(v) = -1$: a subárvore esquerda é mais alta que a direita;
- $fb(v) > 1$: a subárvore direita está desbalanceando o vértice v;
- $fb(v) < -1$: a subárvore esquerda está desbalanceando o vértice v.

Manutenção do balanceamento

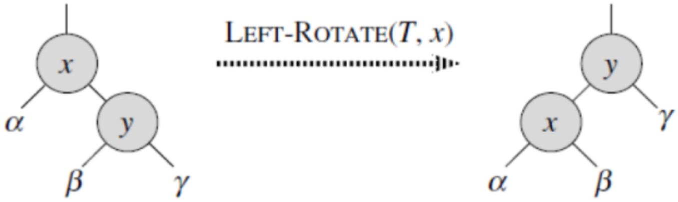
Árvores Binárias de Busca

Balanceamento - Árvores Balanceadas AVL Adelson-Velskii and Landis' Tree

- A manutenção do balanceamento dos vértices em uma árvore AVL é feita sempre que se **insere** ou **remove** um vértice na árvore.
- Nesse sentido, há a necessidade de se transformar a árvore de modo que o percurso (busca) in ordem não seja alterado e que a árvore passe a ser classificada como balanceada.
- Essas transformações são denominadas **rotações**.

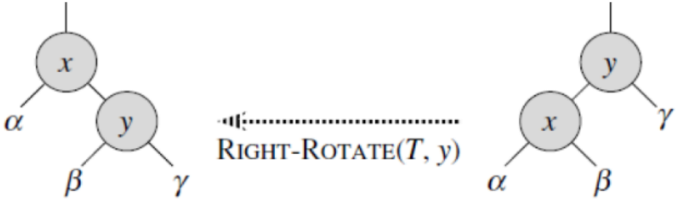
Árvores Binárias de Busca

Balanceamento - Rotação para a esquerda



Árvores Binárias de Busca

Balanceamento - Rotação para a direita



Árvores Binárias de Busca

Balanceamento - Correção de Balanceamento

- Para cada operação (inserção ou remoção), verifica-se cada vértice **ancestral**, até a raiz.
- Caso o vértice v verificado tenha $fb(v) < -1$ ou $fb(v) > 1$, quatro casos (que na verdade são dois casos com simetria) devem ser considerados para efetuar as rotações e garantir o balanceamento.

Árvores Binárias de Busca

Balanceamento - Correção de Balanceamento

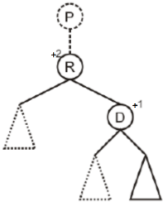
- Seja R o nó desbalanceado, E seu filho esquerdo, D seu filho direito e P seu pai (que pode não existir se R for a raiz da árvore):
  - Caso 1: Se o fator de R é $\geqslant 2$ e o fator de D é $\geqslant 0$, então: Promover left-rotate em R;
  - Caso 2: Se o fator de R é $\geqslant 2$ e o fator de D é < 0, então: Promover right-rotate em D e left-rotate em R
  - Caso 3: Se o fator de R é $\leqslant -2$ e o fator de E é $\leqslant 0$, então: Promover right-rotate em R.
  - Caso 4: Se o fator de R é $\leqslant -2$ e o fator de E é > 0, então: Promover left-rotate em E e right-rotate em R.

Árvores Binárias de Busca

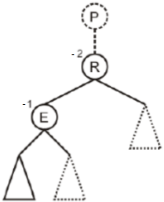
Balanceamento - Correção de Balanceamento

- Caso 1: Se o fator de R é $\geqslant 2$ e o fator de D é $\geqslant 0$, então: Promover left-rotate em R;
- Caso 3: Se o fator de R é $\leqslant -2$ e o fator de E é $\leqslant 0$, então: Promover right-rotate em R.

caso 1



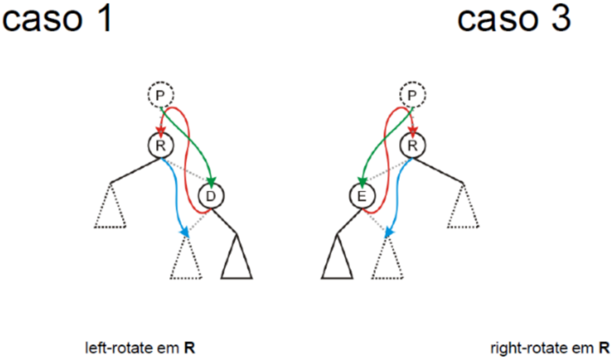
caso 3



Árvores Binárias de Busca

Balanceamento – Correção de Balanceamento

- Caso 1: Se o fator de R é $\geqslant 2$ e o fator de D é $\geqslant 0$, então: Promover left-rotate em R;
- Caso 3: Se o fator de R é $\leqslant -2$ e o fator de E é $\leqslant 0$, então: Promover right-rotate em R.

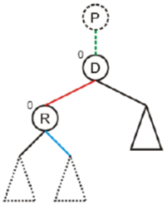


Árvores Binárias de Busca

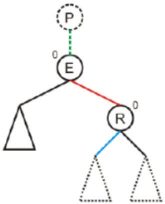
Balanceamento – Correção de Balanceamento

- Caso 1: Se o fator de R é $\geqslant 2$ e o fator de D é $\geqslant 0$, então: Promover left-rotate em R;
- Caso 3: Se o fator de R é $\leqslant -2$ e o fator de E é $\leqslant 0$, então: Promover right-rotate em R.

caso 1



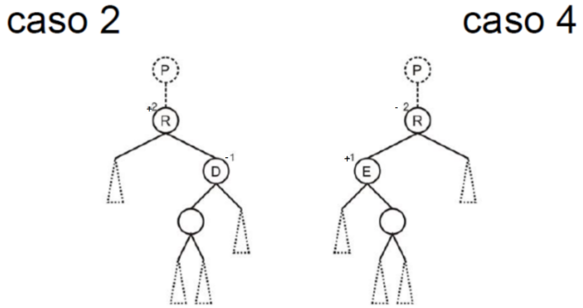
caso 3



Árvores Binárias de Busca

Balanceamento - Correção de Balanceamento

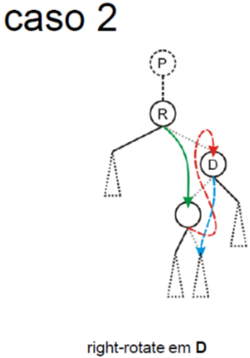
- Caso 2: Se o fator de R é $\geqslant 2$ e o fator de D é < 0, então: Promover right-rotate em D e left-rotate em R
- Caso 4: Se o fator de R é $\leqslant -2$ e o fator de E é > 0, então: Promover left-rotate em E e right-rotate em R.

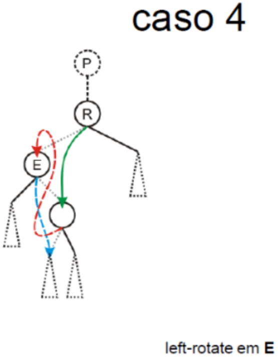


Árvores Binárias de Busca

Balanceamento - Correção de Balanceamento

- Caso 2: Se o fator de R é $\geqslant 2$ e o fator de D é < 0, então: Promover right-rotate em D e left-rotate em R
- Caso 4: Se o fator de R é $\leqslant -2$ e o fator de E é > 0, então: Promover left-rotate em E e right-rotate em R.

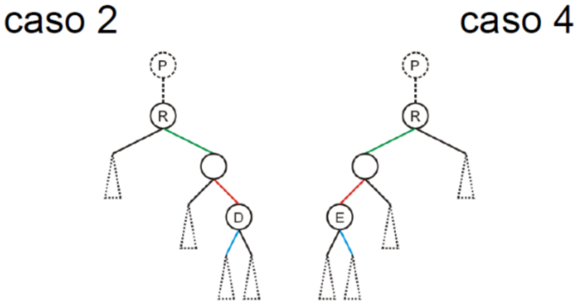




Árvores Binárias de Busca

Balanceamento - Correção de Balanceamento

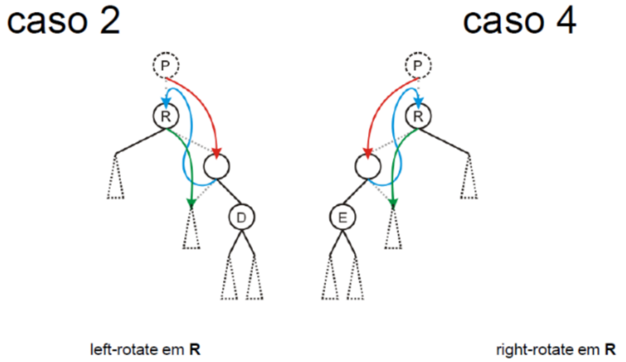
- Caso 2: Se o fator de R é $\geqslant 2$ e o fator de D é < 0, então: Promover right-rotate em D e left-rotate em R
- Caso 4: Se o fator de R é $\leqslant -2$ e o fator de E é > 0, então: Promover left-rotate em E e right-rotate em R.



Árvores Binárias de Busca

Balanceamento - Correção de Balanceamento

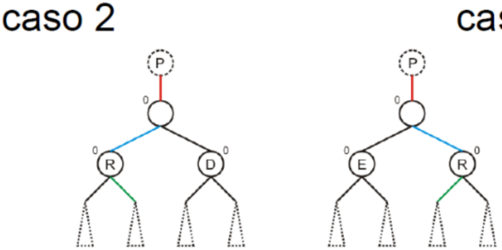
- Caso 2: Se o fator de R é $\geqslant 2$ e o fator de D é < 0, então: Promover right-rotate em D e left-rotate em R
- Caso 4: Se o fator de R é $\leqslant -2$ e o fator de E é > 0, então: Promover left-rotate em E e right-rotate em R.



Árvores Binárias de Busca

Balanceamento – Correção de Balanceamento

- Caso 2: Se o fator de R é $\geqslant 2$ e o fator de D é < 0, então: Promover right-rotate em D e left-rotate em R
- Caso 4: Se o fator de R é $\leqslant -2$ e o fator de E é > 0, então: Promover left-rotate em E e right-rotate em R.



Árvores Binárias de Busca

Exemplo

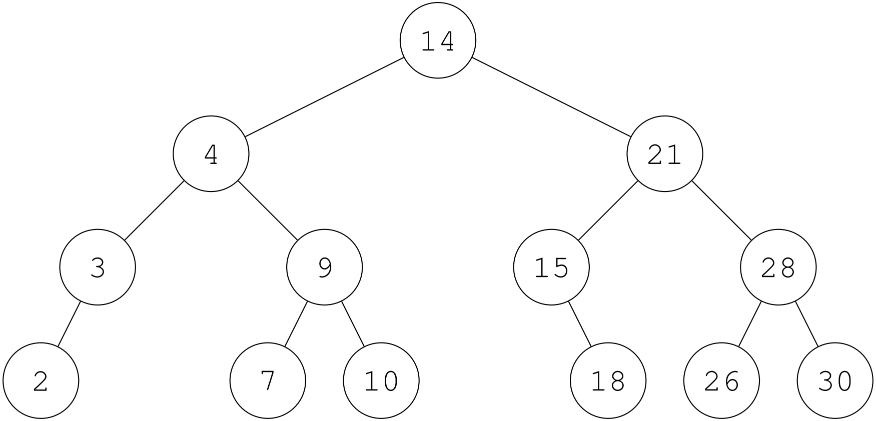
- Faça a inserção dos seguintes valores em uma árvore AVL:

$$21, 26, 30, 9, 4, 14, 28, 18, 15, 10, 2, 3, 7$$

- Lembre-se, para cada valor inserido:
  - Calcule o fator de balanceamento de cada nó;
  - Caso haja desbalanceamento, então faça as rotações necessárias.

Árvores Binárias de Busca

Exemplo



Estrutura de Dados (CCA410)

Aula 06 - Árvores Balanceadas (AVL)

Prof. Luciano Rossi
Prof. Leonardo Anjoletto Ferreira
Prof. Flavio Tonidandel
Prof. Fabio Suim

Ciência da Computação
Centro Universitário FEI

2º Semestre de 2025