Estruturas de Dados II

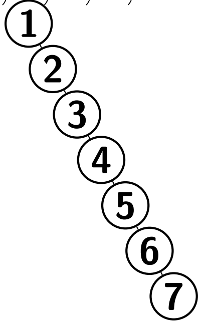
Árvores AVL

Prof[a]. Juliana de Santi
Prof. Rodrigo Minetto
Universidade Tecnológica Federal do Paraná
Material compilado de: Cormen, Notas de aula IC-UNICAMP e IME-USP

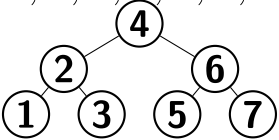
Sumário

Desvantagem de **árvores binárias de pesquisa**: desempenho depende da ordem em que os elementos são inseridos.

1, 2, 3, 4, 5, 6, 7



4, 6, 2, 5, 1, 7, 3



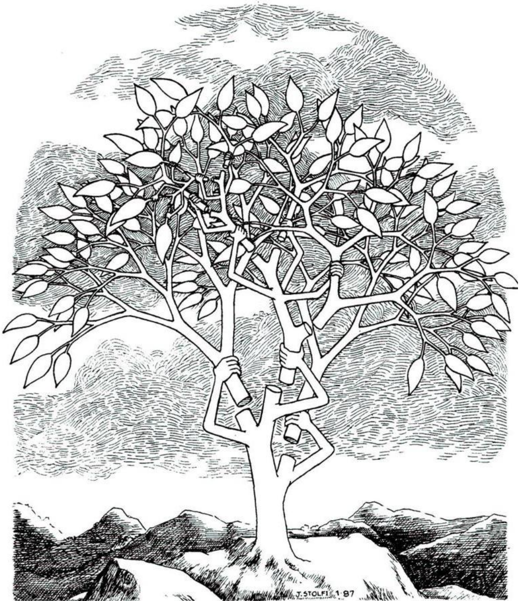
Árvore AVL - Contextualização

O desbalanceamento de uma árvore binária de pesquisa pode tornar uma busca tão ineficiente quanto uma busca sequêncial (no pior caso).

Complexidade: $\mathcal{O}(n)$.

Solução: **árvores balanceadas.**

AVL - Self-adjusting search tree by J. Stolfi (1987)



Árvore AVL

Árvore AVL (ou *árvore balanceada pela altura*): é uma árvore de pesquisa binária auto-balanceada, onde, para cada nó x, as alturas das subárvores esquerda e direita de x diferem de no máximo uma unidade. O nome AVL vem de seus autores **A**delson-**V**elskii e **L**andis, que a descreveram em 1962.

Sumário

Árvore AVL - Fator de balanceamento

Por questões de otimização, em uma árvore AVL, todo **nó** armazena a sua altura, esse campo é utilizado **para calcular** o **fator de balanceamento**, que indica a situação de equilíbrio do nó:

$$f_b(\mathrm{x}) = h_d(x) \text{ - } h_e(x)$$

tal que h_d se refere a **altura da árvore direita** e h_e a **altura da árvore esquerda** do nó x.

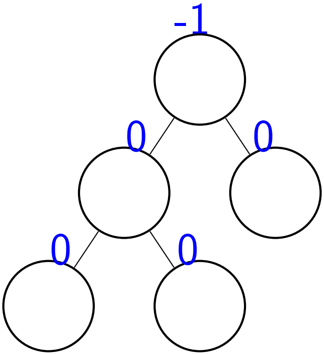
Árvore AVL - Fator de balanceamento

Cada nó **x** numa árvore AVL deve ter um f_b de **-1**, **0** ou **1**. Tal que:

- **-1** sub-árvore esquerda de **x** tem um nível a mais que a da direita.
- **0** sub-árvores de **x** equilibradas.
- **1** sub-árvore direita de **x** tem um nível a mais que a da esquerda.

Árvore AVL - Exemplo

Exemplo de árvore AVL.



Árvore AVL - Fator de balanceamento

No entanto, as operações de **inserção** e/ou **remoção** de nós em uma árvore AVL não garantem que a árvore permanecerá balanceada. Ou seja, o **fator de balanceamento** de -1, 0 ou 1 pode ser **violado**.

Sumário

Árvore AVL - Rotação

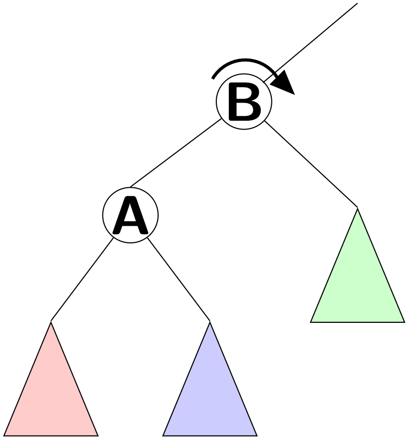
Para manter o balanceamento em uma árvore AVL é necessário fazer uma **transformação** tal que:

- a árvore transformada permaneça balanceada.
- a árvore continue sendo binária de busca.

Esta transformação é conhecida como **rotação** (cujo efeito é o de rearranjo dos nós da árvore).

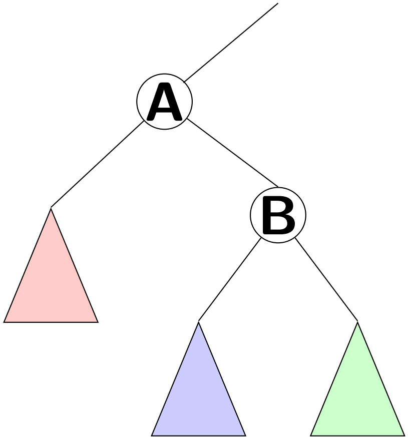
Árvore AVL - Rotação

Existem duas operações básicas para balancear uma árvore AVL: **rotação à direita.**

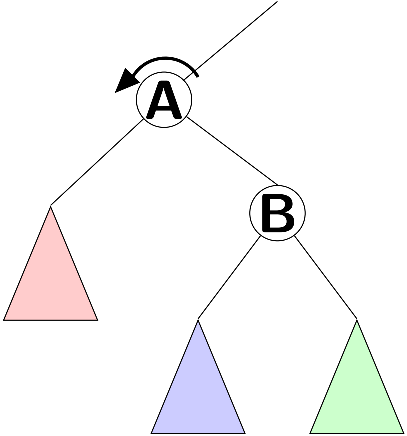


Árvore AVL - Rotação

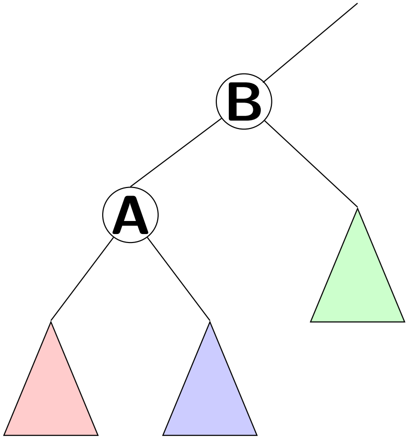
Existem duas operações básicas para balancear uma árvore AVL:



Existem duas operações básicas para balancear uma árvore AVL: **rotação à esquerda.**



Existem duas operações básicas para balancear uma árvore AVL:



Dependendo do desbalanceamento a ser solucionado, apenas uma rotação não será suficiente para resolvê-lo. Solução: **rotação dupla**.

Sumário

Árvore AVL - Inserção

O que acontece quando um novo nó é **inserido**? Suponha um nó x, com subárvores h_e e h_d e uma **inserção** que **aumenta** h_e.

- Se $h_e = h_d$, então h_e e h_d ficam com alturas diferentes mas balanceados.
- Se $h_e < h_d$, então h_e e h_d ficam com alturas iguais e balanceamento foi melhorado.
- Se $h_e > h_d$, então h_e fica ainda maior e o balanceamento foi **violado**.

Árvore AVL - Inserção

O sinal do fator de balanceamento indica qual é o tipo de rotação que deve efetuada a fim de rebalancear a árvore.

- Se o f_b é **negativo**, as **rotações** são feitas à **direita**.
- Se o f_b é **positivo**, as **rotações** são feitas à **esquerda**.

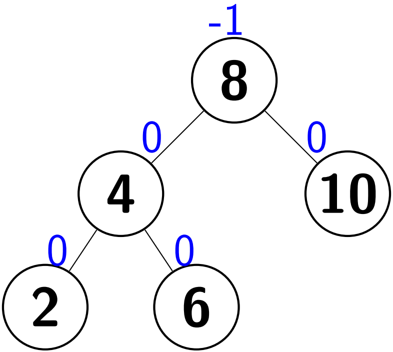
Árvore AVL - Inserção

Há duas situações nos casos de rebalanceamento:

Caso 1: nó com $f_b = 2$ (ou -2) e filho (na direção da inserção) com $f_b = 1$ (ou -1), ou seja, fator de balanceamento f_b de **mesmo sinal**.

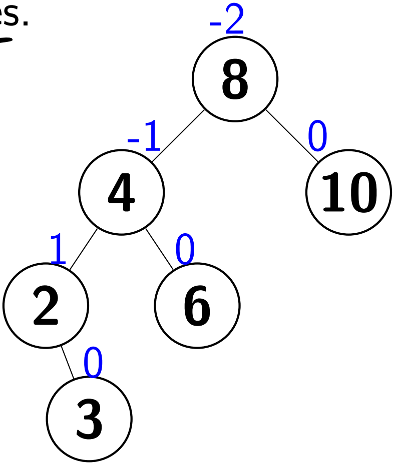
Solução: **rotação simples!**

Dada uma árvore AVL, tal como abaixo, suponha a inserção do valor 3.

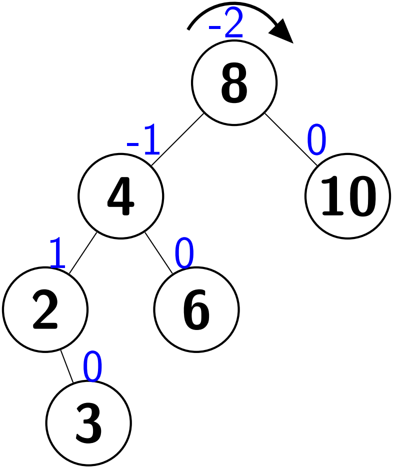


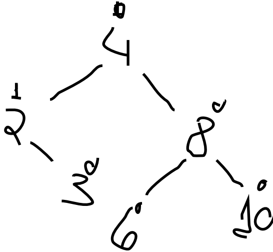
Árvore AVL - Inserção

A raiz (nó 8) ficou desbalanceada $f_b = -2$, e os sinais dos f_b dos nós 8 e 4 são iguais. Solução: rotação simples.

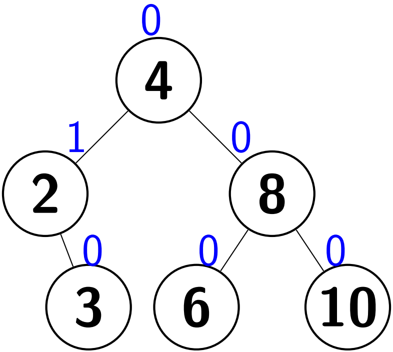


Note que o f_b do nó 8 é negativo então realizamos uma rotação à direita.





Resultado: **árvore rebalanceada!**

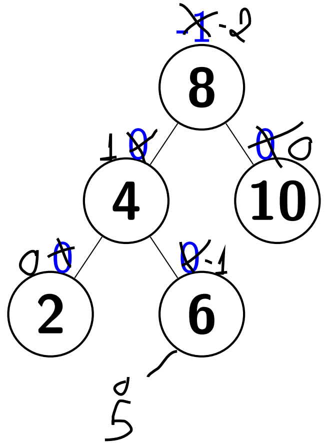


Caso 2: nó com $f_b = 2$ (ou -2) e filho (na direção da inserção) com $f_b = -1$ (ou 1), ou seja, fator de balanceamento f_b com **sinal diferente**.

Solução: **rotação dupla!** Primeiro rotaciona-se o filho e depois o nó com fator de balanceamento de 2 (ou -2).

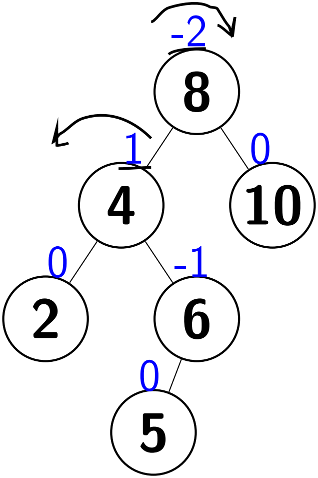
Árvore AVL - Inserção

Dada uma árvore AVL, tal como abaixo, suponha a inserção do valor 5.



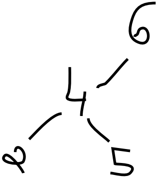
Árvore AVL - Inserção

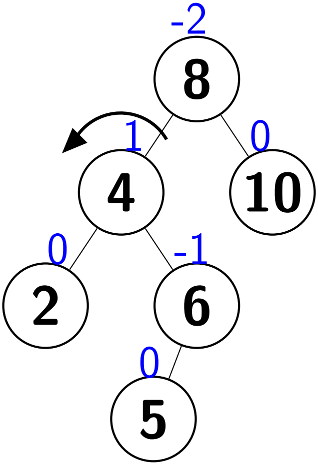
A raiz (nó 8) ficou desbalanceada $f_b = -2$, e os sinais dos f_b dos nós 8 e 4 são diferentes. Solução: rotação dupla.



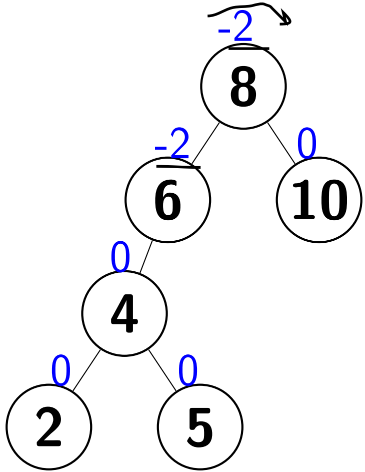
Árvore AVL - Inserção

Primeiro rotaciona-se o filho (nó 4). Como o f_b do nó 4 é positivo então realizamos uma rotação à esquerda.



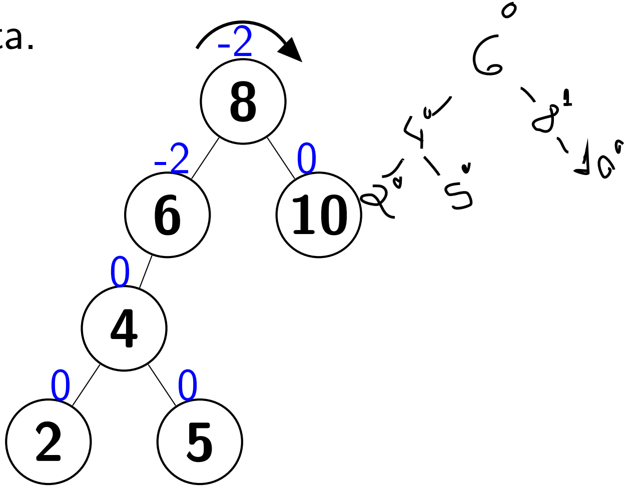


Note que ambos os nós (8 e o seu novo filho 6) agora têm fatores de balanceamento f_b com o mesmo sinal.

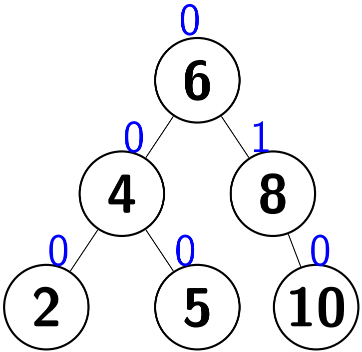


Árvore AVL - Inserção

Posteriormente, rotaciona-se o pai (nó 8). Como o f_b do nó 8 é negativo então realizamos uma rotação à direita.



Resultado: **árvore rebalanceada!**



Árvore AVL - Inserção

Nota: em uma operação de **inserção**, sempre é possível restaurar o balanceamento com no máximo **uma operação**, seja ela uma rotação simples ou dupla.

Sumário

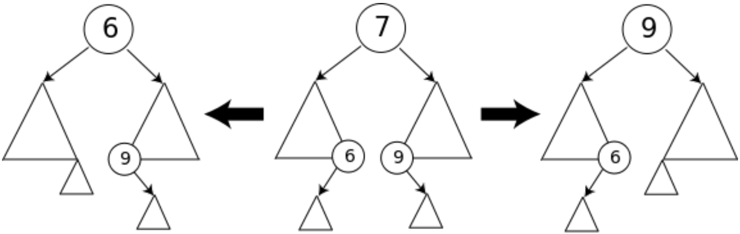
Árvore AVL - Remoção

A remoção de um nó em uma árvore AVL pode ser dividida em três casos, podendo em qualquer um deles ser necessário rebalancear a árvore através de rotações.

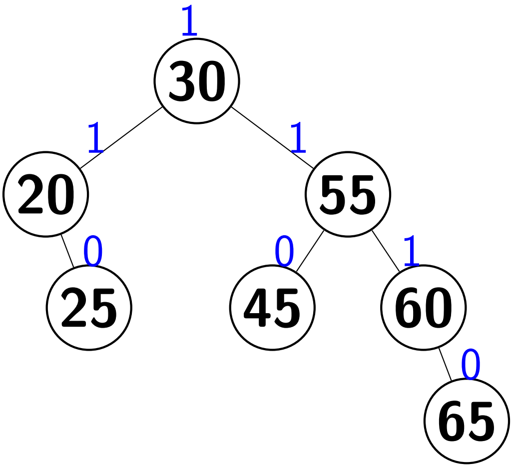
Caso 1 - o nó é folha: remova-o.

Caso 2 - o nó tem apenas um filho: remova-o e faça o filho assumir o lugar do pai.

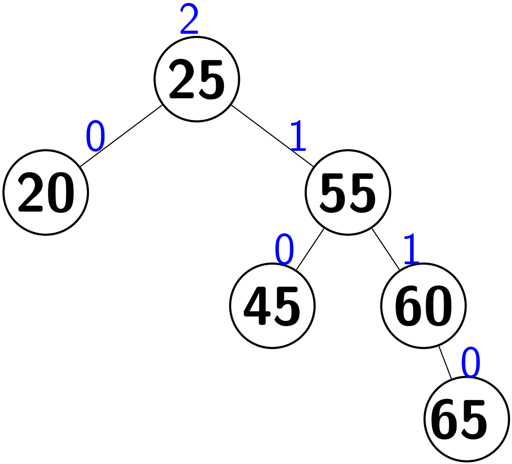
Caso 3 - o nó tem descendentes à esquerda e/ou direita: remova-o e promova o maior elemento da sub-árvore à esquerda (ou o menor da sub-árvore direita) ao seu lugar. Exemplo: remoção do nó 7 (centro):



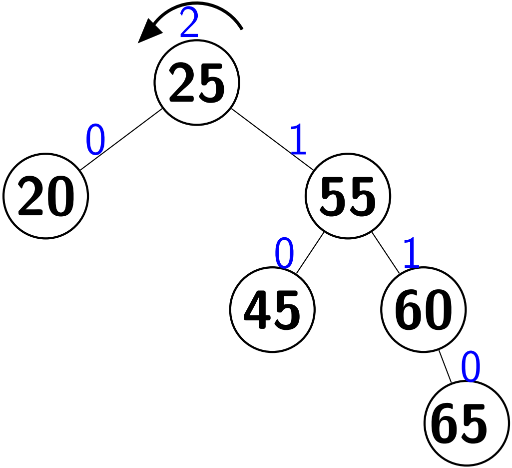
Rotação simples: dada a árvore AVL abaixo, remova o valor 30.



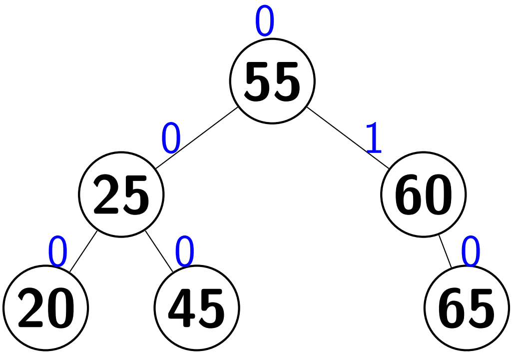
Note que a nova raiz (nó 25) ficou desbalanceada $f_b = +2$.



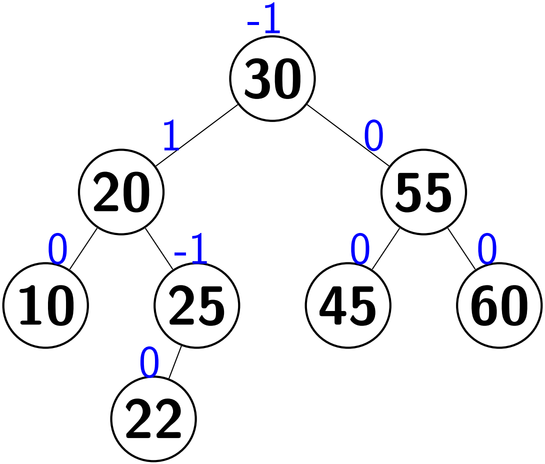
Como o f_b do nó 25 é positivo é suficiente uma rotação à esquerda.



Resultado: **árvore rebalanceada!**

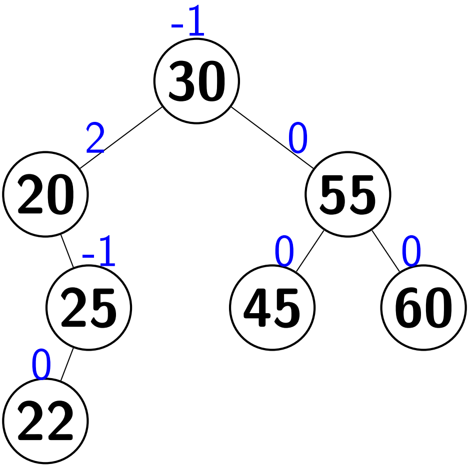


Rotação dupla: dada a árvore AVL abaixo, remova o valor 10.



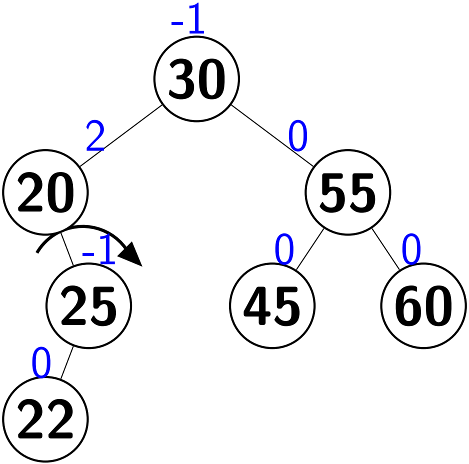
Árvore AVL - Remoção

Note que o nó 20 ficou desbalanceado $f_b = +2$.



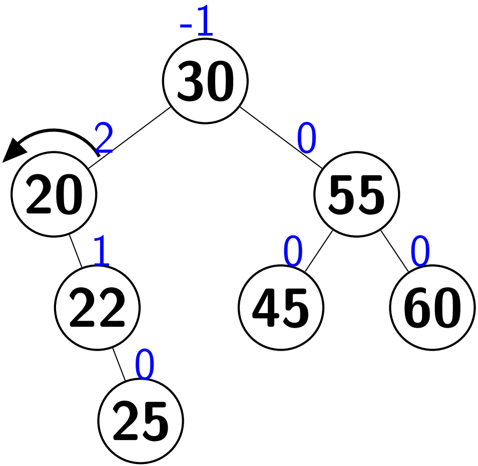
Árvore AVL - Remoção

Como o f_b do nó 25 é negativo é necessário uma rotação dupla.

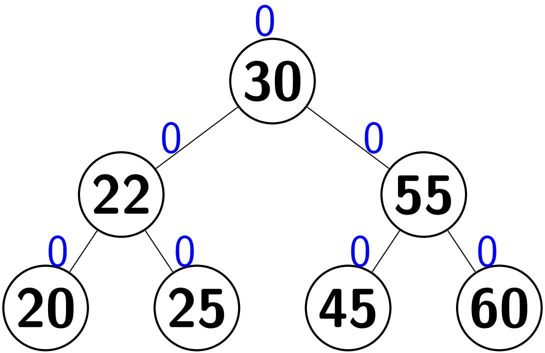


Árvore AVL - Remoção

E finalmente é rotacionado à esquerda o nó 20.



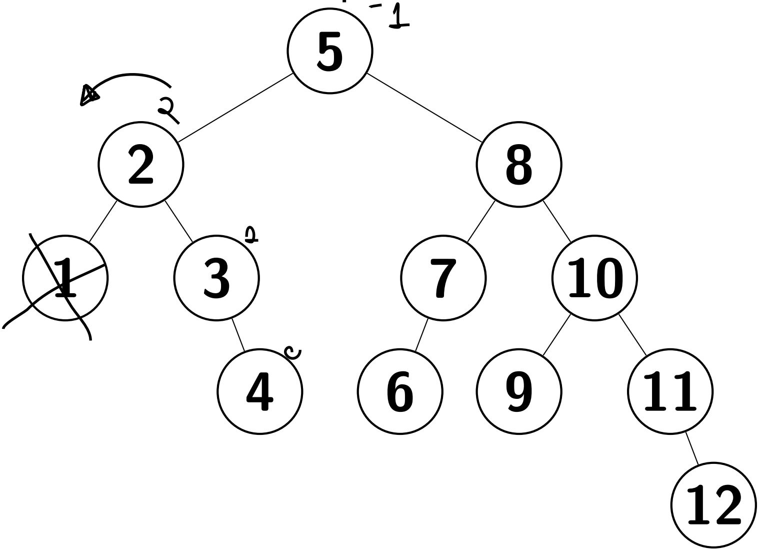
Resultado: **árvore rebalanceada!**



Nota: em uma operação de remoção pode haver a necessidade de realizar mais de duas rotações (o que não acontece na inserção), podendo se estender para uma rotação em cada nível.

Árvore AVL - Remoção

Remova o valor 1. O que acontece?



Sumário

Árvore AVL - Complexidade

Rotação: $\mathcal{O}(1)$ (para realizar uma rotação é necessário re-atribuir os elementos (ponteiros) da esquerda e direita de alguns poucos nós).

Busca: $\mathcal{O}(\log n)$ (n é o número de nós na AVL).

Inserção: $\mathcal{O}(\log n)$ (busca + rotação)

Remoção: $\mathcal{O}(\log n)$ (busca + rotações)

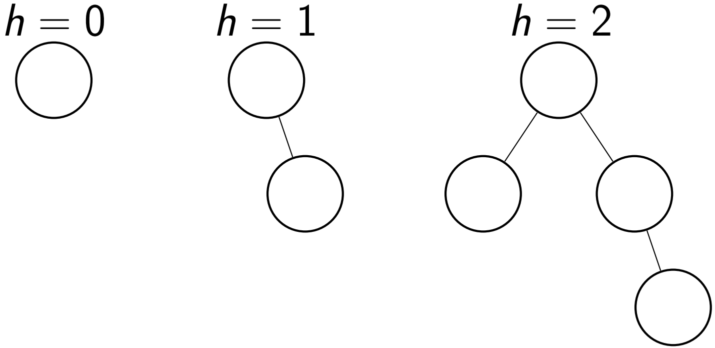
Árvore AVL - Altura

O custo (complexidade) das operações em uma árvore AVL é definido pela altura máxima h de uma árvore AVL com n nós.

Pergunta: qual a altura máxima h de uma árvore com n nós?

Árvore AVL - Altura

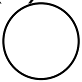
Prova: seja uma árvore AVL esparsa de altura h formada com o número mínimo de nós. Exemplos de árvores:



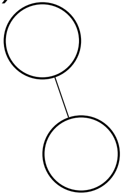
Árvore AVL - Altura

Seja $S(h)$ uma função que devolve o número de nós de uma árvore AVL esparsa de altura h. Por exemplo:

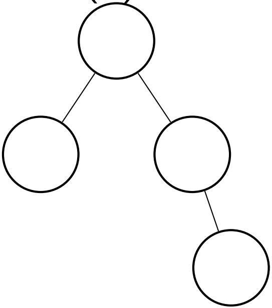
$S(0) = 1$



$S(1) = 2$

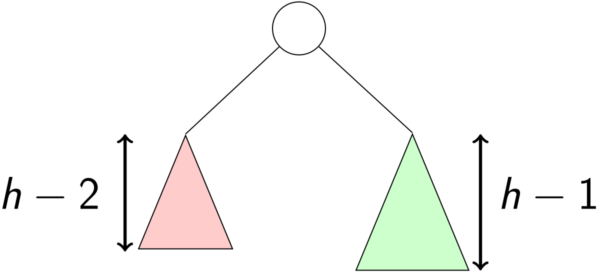


$S(2) = 4$



Árvore AVL - Altura

Uma árvore AVL esparsa de altura h, para $h \geq 1$, pode ser construída a partir de árvores AVL esparsas de alturas $h - 1$ e $h - 2$.



Logo, temos que

$$S(h) = S(h-2) + S(h-1) + 1$$

Ou seja, o número total de nós da árvore com altura h é a soma dos nós de suas subárvores esquerda e direita mais um elemento que se refere ao nó raiz que conecta as subárvores.

Árvore AVL - Altura

$$
\begin{aligned}
\text{Logo: } S(h) &= S(h-2) + S(h-1) + 1 \\
&\geq 1 + 2S(h-2) \\
&\qquad \text{pois S(h-2) é a árvore menor} \\
&\geq 1 + 2(1 + 2S(h-4)) \\
&\geq 1 + 2 + 2^2 S(h-4) \\
&\geq 1 + 2 + 2^2 + 2^3 S(h-6) \\
&\dots \\
&\geq 1 + 2 + 2^2 + 2^3 + \dots + 2^{h/2} \\
&\quad 2^{h/2} \text{ pois a cada passo examinamos} \\
&\qquad \text{1 subárvore.} \\
&\geq 2^{h/2-1}
\end{aligned}
$$

Árvore AVL - Altura

Sabendo que para qualquer árvore AVL com n nós $2^{h/2-1} \leq S(h) \leq n$. Então

$$\begin{aligned}
2^{h/2-1} &\leq n \\
\log_2 2^{h/2-1} &\leq \log_2 n \\
h/2 - 1 &\leq \log_2 n \\
h &\leq 2\log_2 n + 2 \\
h &\in \mathrm{O}(\log n)
\end{aligned}$$

Portanto fica **provado** que uma árvore AVL com n nós não tem altura maior que $\mathrm{O}(\log n)$.

Referências

[1] http://courses.csail.mit.edu/6.006/spring11/rec/rec04.pdf

[2] http://www-cs-faculty.stanford.edu/~eroberts/courses/cs106b/chapters/13

[3] http://www.lcad.icmc.usp.br/~nonato/ED/AVL/remocao.html

[4] http://www.cse.ohio-state.edu/~gurari/course/cis680/cis680Ch10.html