

Estrutura de Dados II



Unidade 5 – Árvores AVL

Prof. Sandro T. Pinto



Árvore AVL é uma árvore binária de busca balanceada, ou seja, uma árvore balanceada são as árvores que minimizam o número de comparações efetuadas no pior caso para uma busca com chaves de probabilidades de ocorrências idênticas.

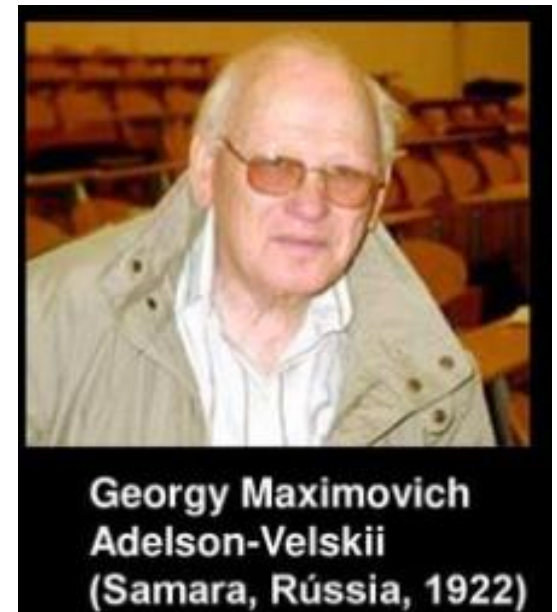
Wikipédia (2021)



Unidade 5

Árvores AVL

O nome **AVL** vem de seus criadores soviéticos Adelson, Velsky e Landis, e sua primeira referência encontra-se no documento "Algoritmos para organização da informação" de 1962.



Unidade 5

Árvores AVL

Características:

- Manter uma árvore binária de busca balanceada sob a presença de constantes inserções e deleções é **ineficiente**.
 - Para contornar esse problema foi criada a árvore AVL.
- A árvore AVL é uma árvore binária com uma **condição de balanço**, porém não completamente balanceada.
- As árvores AVL permitem inserção / deleção e **rebalanceamento** aceitavelmente rápidos.

Unidade 5

Árvores AVL

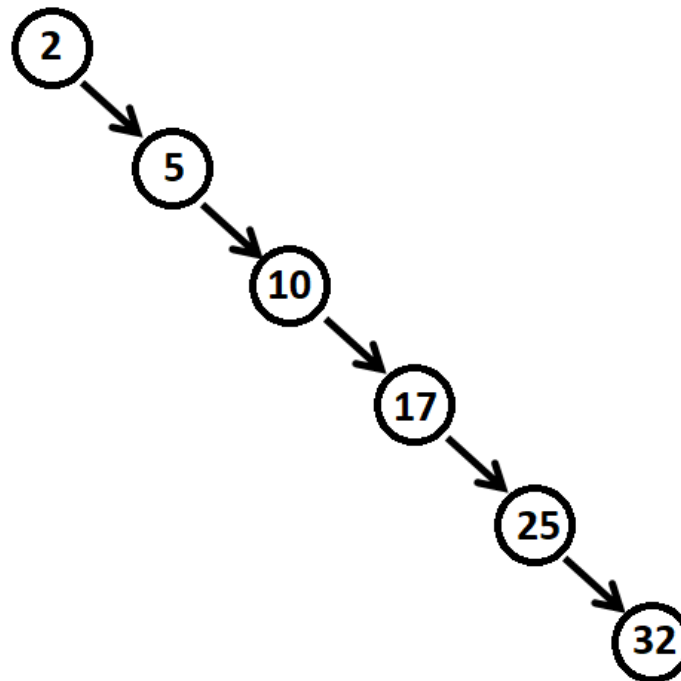
A forma como os elementos são inseridos em uma árvore binária de busca pode fazer com que a busca se torne altamente ineficiente. Considere o conjunto de dados:

2 - 5 - 10 - 17 - 25 - 32

Unidade 5

Árvores AVL

Se inserirmos tais dados, do primeiro até o último, em uma árvore de busca binária, teremos a árvore estruturada da seguinte forma:



Unidade 5

Árvores AVL

Repare que a estrutura de dados se assemelha muito mais a uma lista linear encadeada, do que com uma árvore binária de busca, desta forma denominamos como árvore desbalanceada.

A partir disso, surge o conceito de **balanceamento**.

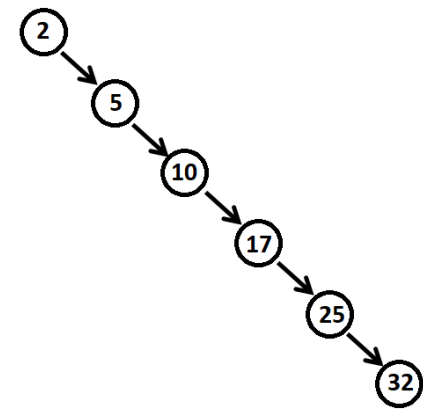
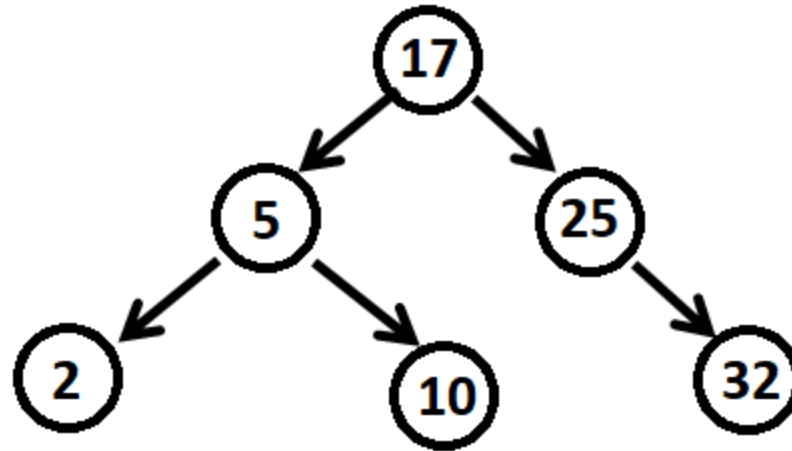


Unidade 5

Árvores AVL

Uma árvore balanceada tende a manter sua altura tão pequena quanto possível, à medida em que são realizadas novas inserções ou remoções de dados.

Considerando o mesmo conjunto de dados, podemos tentar balancear a árvore de busca binária, ficando da seguinte forma:

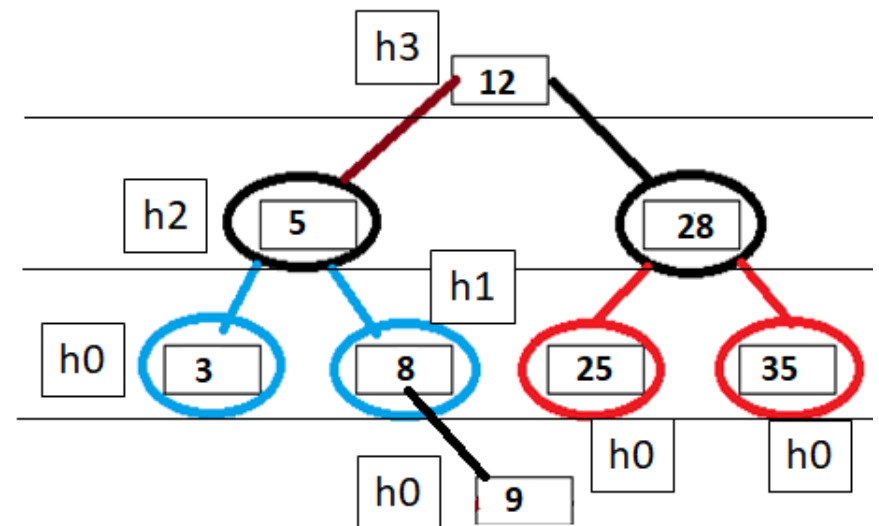


Unidade 5

Árvores AVL

O conceito de balanceamento está relacionado à **altura das sub árvores** que compõe uma árvore binária.

É importante lembrar que a altura (ou profundidade) de uma sub árvore é igual ao número de nós visitados desde a raiz até o nó folha mais distante.



Unidade 5

Árvores AVL

Dizemos que um nó está balanceado caso o valor absoluto da diferença entre as alturas das sub árvores esquerda e direita seja menor ou igual a 1.

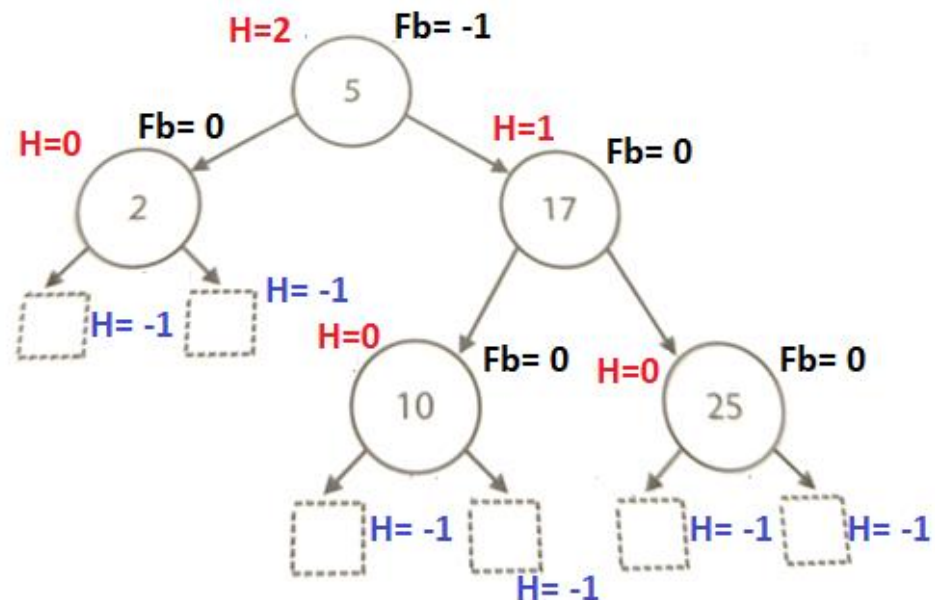
Denominamos de **fator de balanceamento** (Fb) o resultado da diferença entre a altura da **sub árvore esquerda** (He) de um nó pela altura da **sub árvore direita** (Hd) do mesmo nó, de acordo com a seguinte fórmula:

$$Fb = He - Hd$$

Unidade 5

Árvores AVL

Essa árvore encontra-se balanceada, porque a **altura H** de cada sub árvore encontra-se na porção **superior esquerda** de cada nó, e o respectivo fator de **balanceamento Fb** no canto **superior direito** de cada nó.

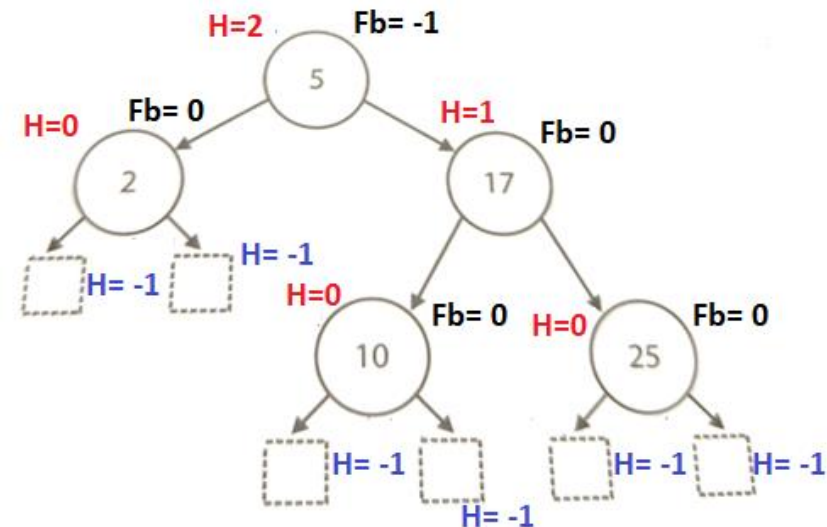


Unidade 5

Árvores AVL

Além disso, as raízes das sub árvores vazias estão destacadas como quadrados pontilhados, que apresentam altura -1. Repare como cada fator de balanceamento (destacados ao lado direito de cada nó), é o resultado do valor da altura da sub árvore esquerda menos a altura da sub árvore direita.

$$Fb = H_e - H_d$$



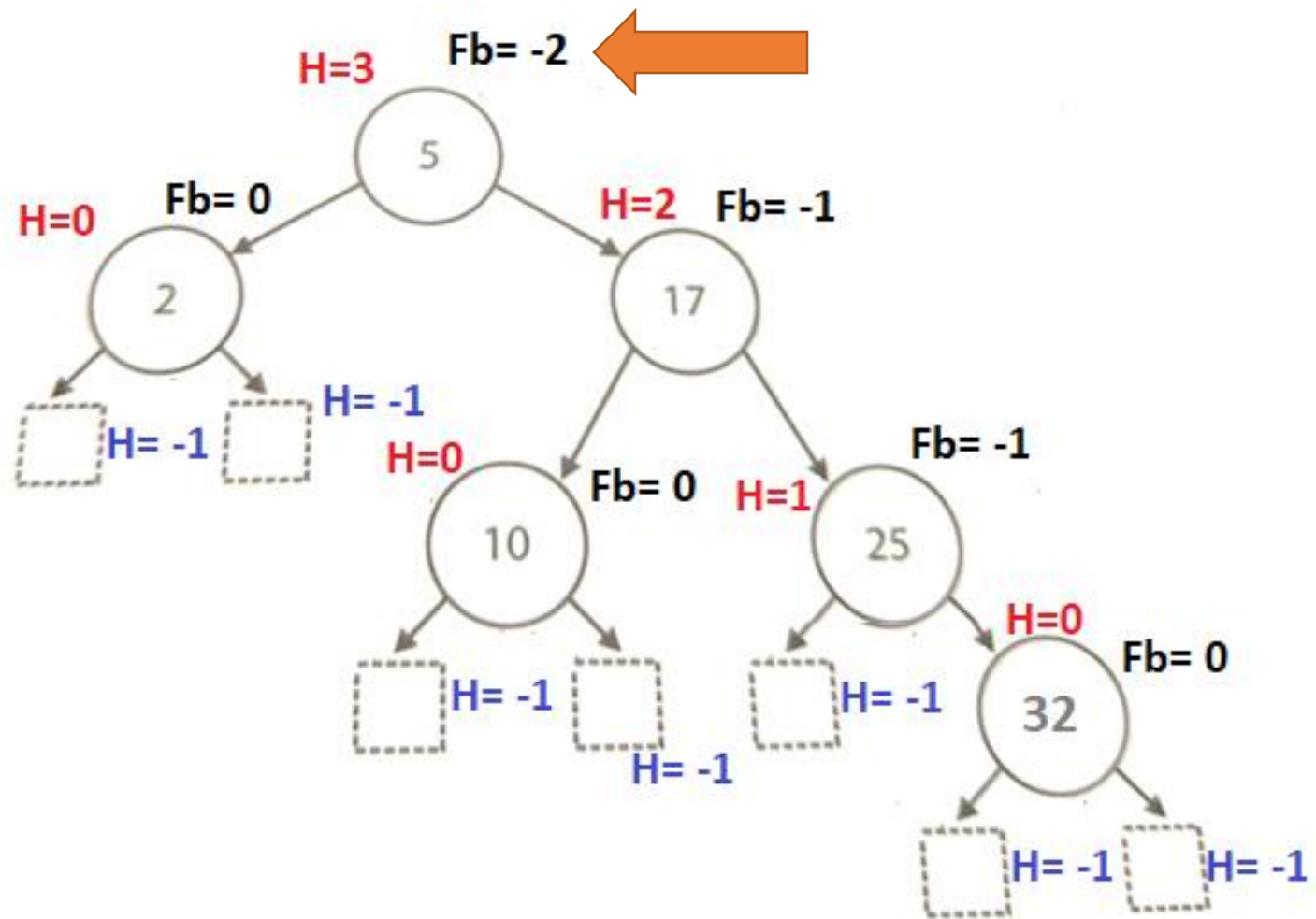
Unidade 5

Árvores AVL

Todavia existem situações nas quais uma árvore binária possui nós desbalanceados . Por exemplo, imagine que adicionemos um nó cujo o valor é igual a 32. Segundo as regras de inserção para árvore binária de busca, tal nó seria adicionado como filho à direita do nó de valor 25. veja:

Unidade 5

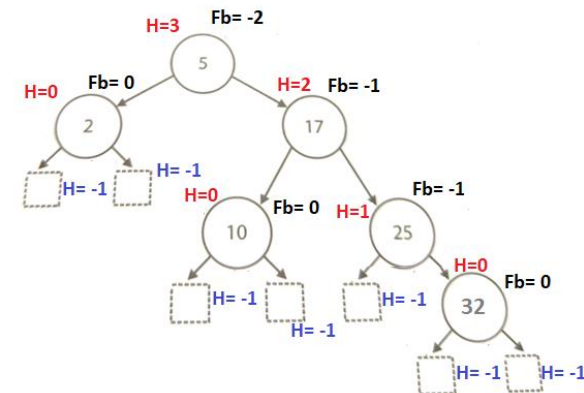
Árvores AVL



Unidade 5

Árvores AVL

Analisando a Figura, é possível notar que a inserção do nó de valor 32 fez com que a raiz da árvore perdesse seu balanceamento. Note que a altura da sub árvore esquerda, em relação à raiz, é igual a 0, porém a altura da sub árvore direita é igual a 2. Com isso, o fator de balanceamento da raiz é igual a -2 que, em valor absoluto, é maior do que 1. Sendo assim o nó 5 encontra-se desbalanceado.



Unidade 5

Árvores AVL

A fim de resolver o problema do desbalanceamento de ABB, os pesquisadores **Adelson – Velskii – Landis**, em 1962, criaram um algoritmo que leva as iniciais de seus nomes. A árvore **AVL** são, nesse sentido, árvores nas quais todos os nós encontram-se balanceados. Para cada nó, a diferença entre as alturas de suas **sub - árvores não podem ser igual ou superior a 2**.

Unidade 5

Árvores AVL

Uma árvore binária balanceada pode **perder** essa característica quando um elemento é inserido ou removido.

Desta forma, quando ocorrem operações de inserção ou remoção em árvores AVL, recalcula-se os fatores de balanceamento de cada nó para, assim, poder executar rotações nos nós problemáticos, **na tentativa de restabelecer seu balanceamento.**

Unidade 5

Árvores AVL

Existem quatro tipos de rotações:

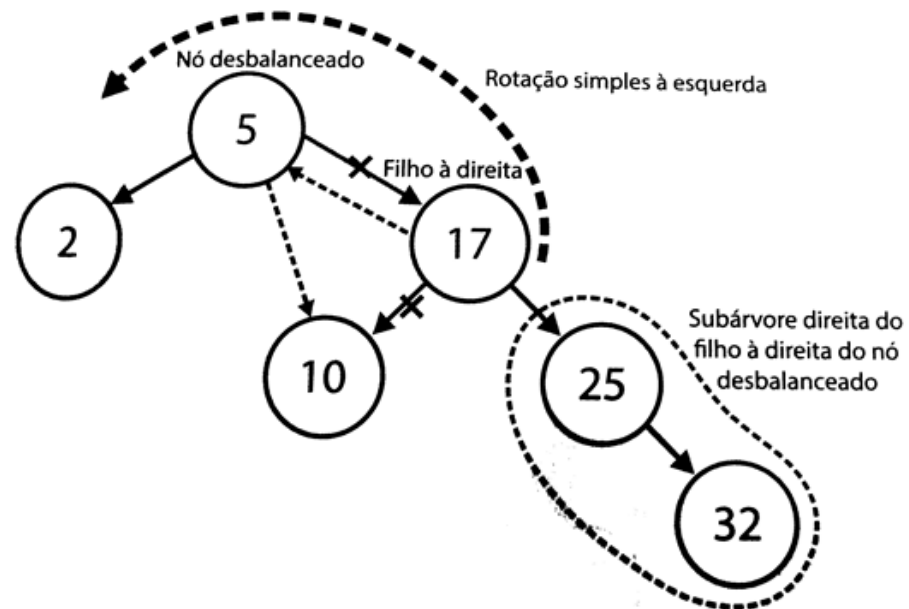
- 2 rotações simples;
- 2 rotações duplas.

Quando uma árvore AVL fica desbalanceada após a inserção de um nó na sub árvore que se encontra à **direita do filho direito** ou, um nó na sub árvore **esquerda do filho da esquerda**, é preciso realizar um **rotação simples**.

Unidade 5

Árvores AVL

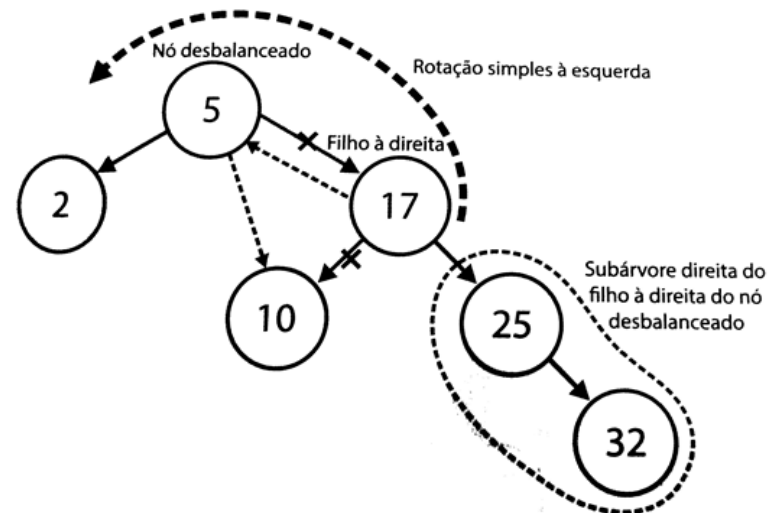
Foi realizada uma inserção na sub árvore direita do filho direito em relação ao nó desbalanceado. Ou seja, o **nó 32** foi inserido na sub árvore direita do **nó 17**, que é filho à direita do **nó 5**. Para tornar a árvore balanceada, precisamos realizar uma rotação, veja:



Unidade 5

Árvores AVL

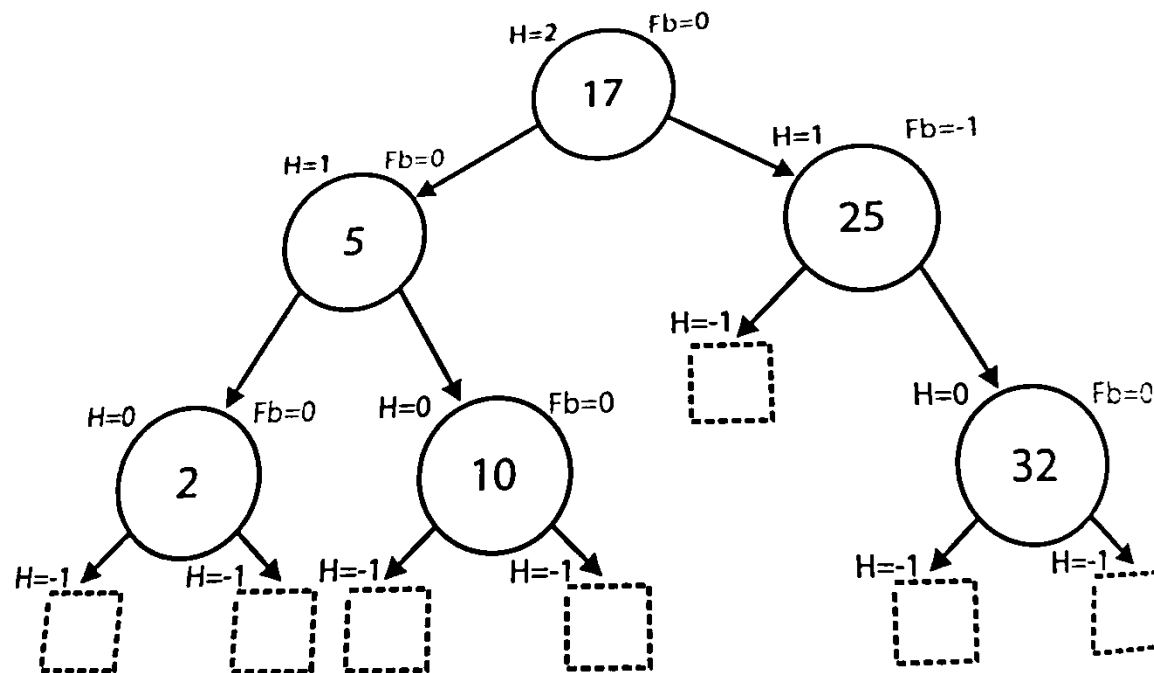
Para realizarmos a rotação à esquerda, precisamos “puxar” o **nó 17** para cima, fazendo com que o **nó 5** passe a ser filho à esquerda do **nó 17**. Além disso, a sub árvore com raiz no **nó 10** é “deserdada” pelo **nó 17**, e o **nó 5** a “adota”, fazendo com que o **nó 10** passe a ser filho à direita do **nó 5**.



Unidade 5

Árvores AVL

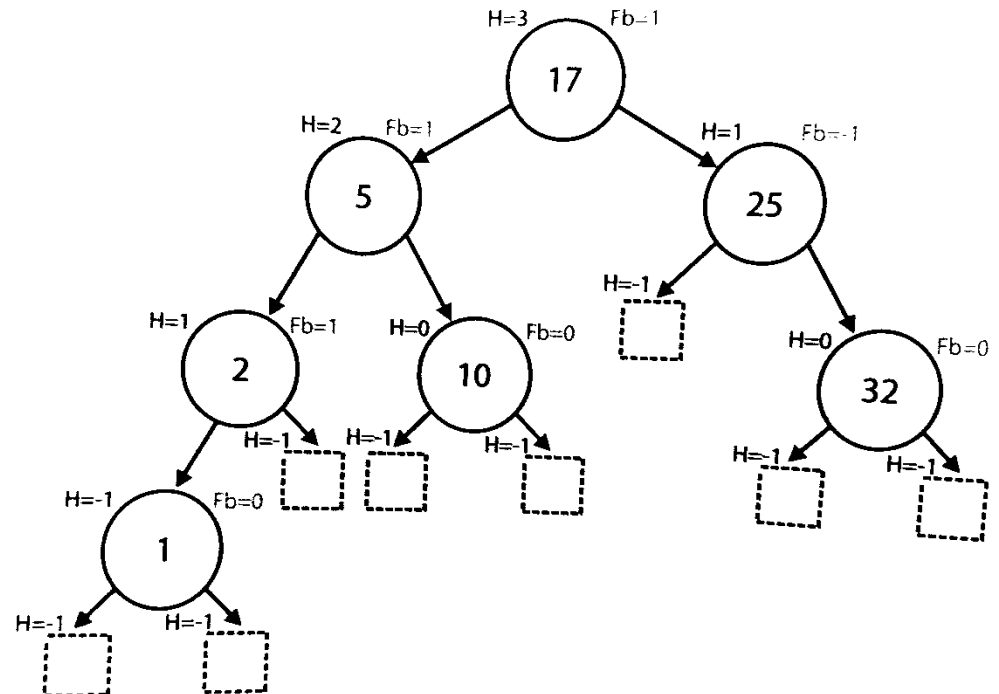
Após realizada a rotação, devemos **recalcular a altura** de cada sub árvore, bem como os fatores de balanceamento de cada nó.



Unidade 5

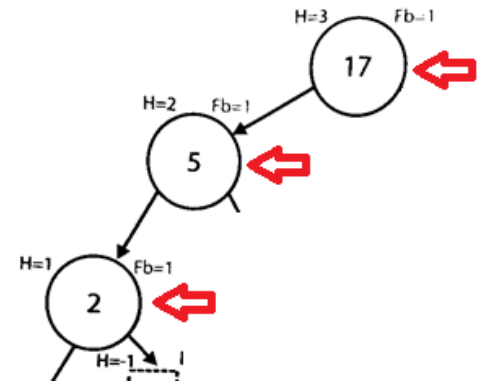
Árvores AVL

A partir dessa nova árvore balanceada, adicionamos um **nó de valor igual a 1**. Para isso, devemos continuar seguindo a regra de inserção em uma ABB. Diante disso, inicia-se pela raiz 17 em direção a uma folha até inserir o 1 à esquerda do nó 2.



Unidade 5

Árvores AVL



Sempre que inserirmos um nó, devemos checar se todos os nós percorridos desde a raiz até o nó folha estão balanceados.

É possível reparar que, após a inserção, foram **recalculados os fatores de balanceamento Fb** dos nós 17, 5 e 2 e nenhum deles se tornou desbalanceado.

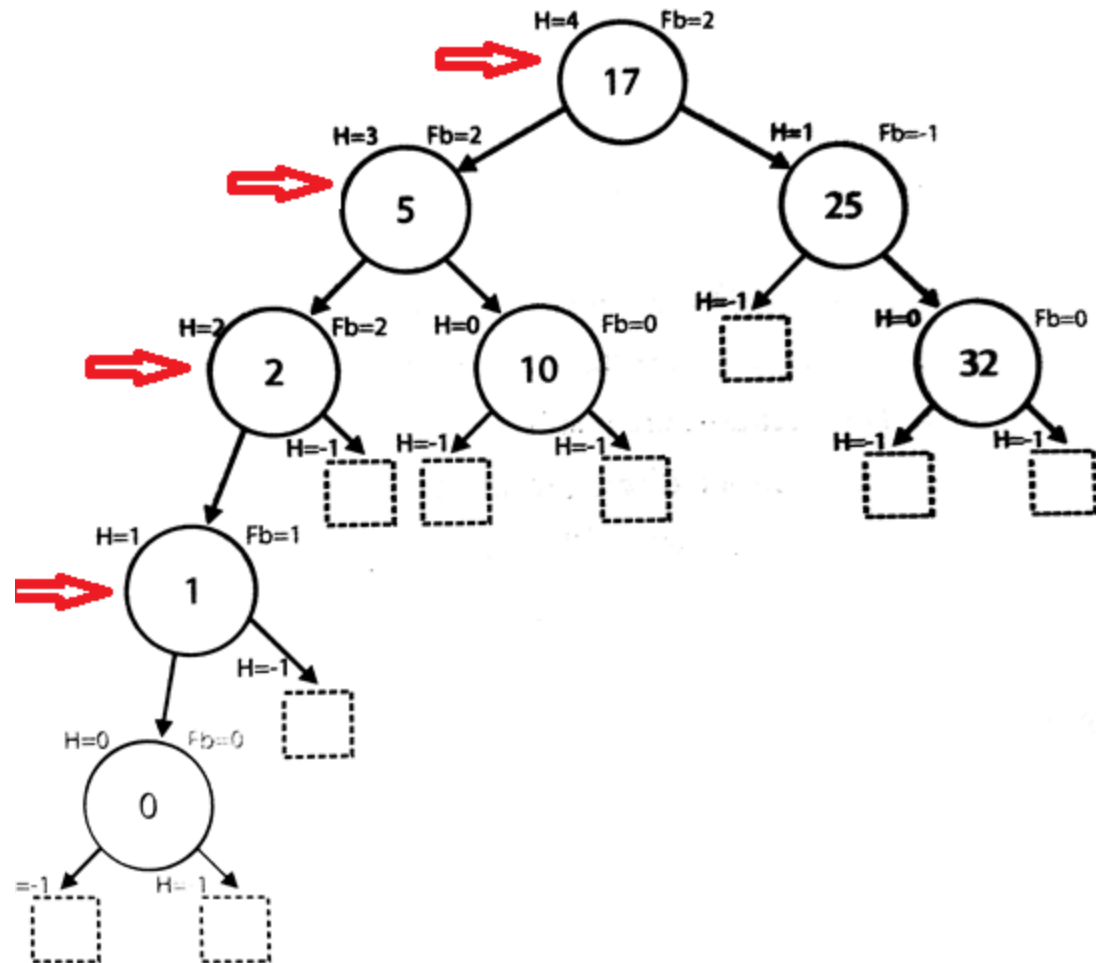
Repare também que somente os nós 17, 5 e 2 tiveram seus **fatores de balanceamento alterados**.

Sendo assim reforça o fato que **apenas os nós percorridos** devem ser checados à procura de desbalanceamento.

Unidade 5

Árvores AVL

Agora vamos inserir um nó com valor 0. Para adicioná-lo tivemos que percorrer os nós 17, 5, 2 e 1 para finalmente adicionar o nó 0 à esquerda do nó 1.



Unidade 5

Árvores AVL

Ao recalcular os fatores de balanceamento, chegamos à conclusão de que os nós 2, 5 e 17 estão desbalanceados.

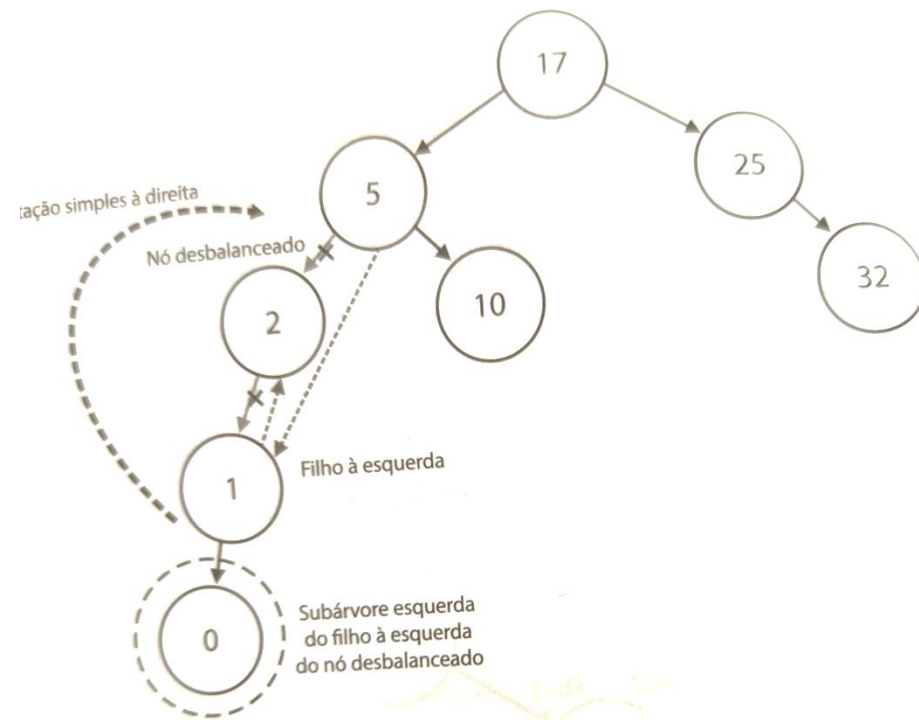
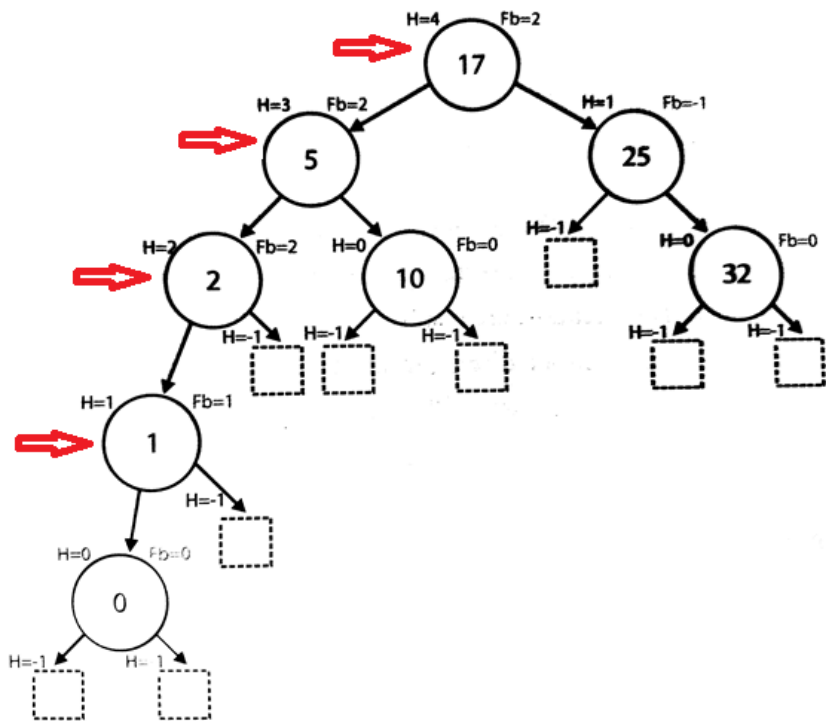
A partir dessas informações temos condições de tentar consertar nossa árvore, de baixo para cima.

OBS: é importante ressaltar que, apenas os nós percorridos durante a inserção tiveram chance de **se tornar desbalanceados**, pois somente suas sub árvores tiveram as respectivas **alturas alteradas na inserção**.

Unidade 5

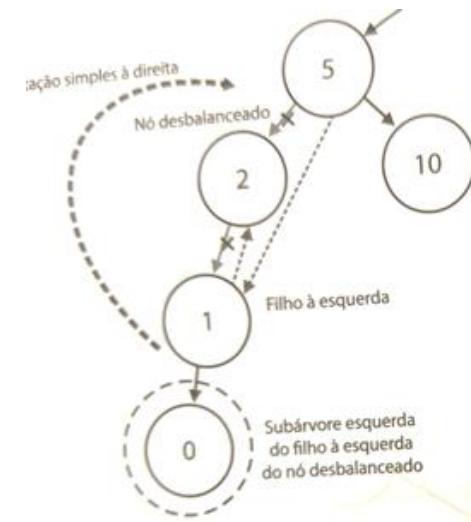
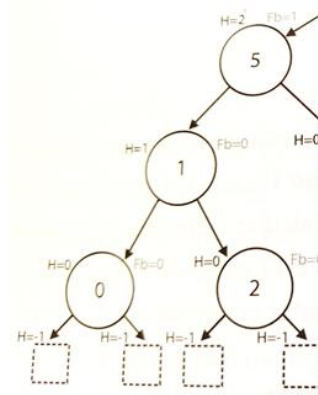
Árvores AVL

Corrigindo uma inserção na sub árvore esquerda do filho à esquerda com rotação simples a direita.



Unidade 5

Árvores AVL



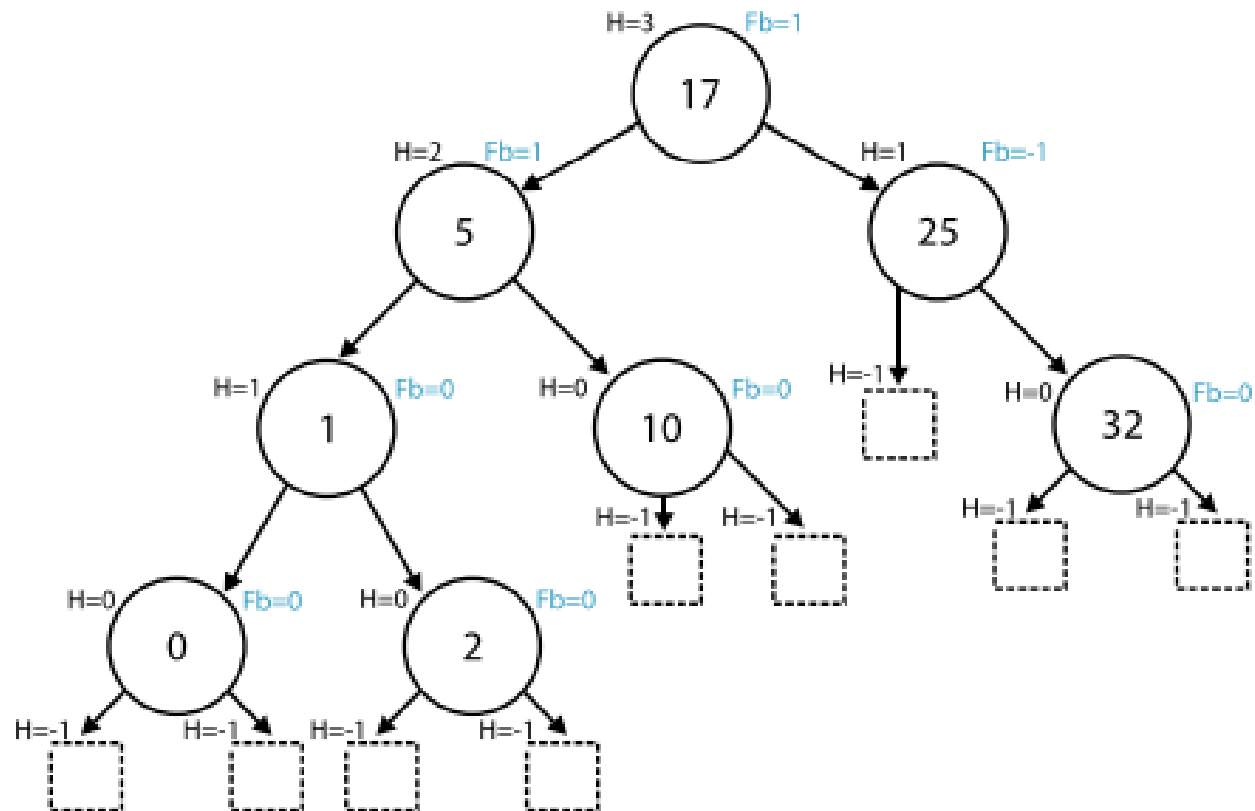
De maneira simétrica à situação que levou à rotação feita anteriormente, temos o caso oposto.

Agora realizamos a inserção de um nó na sub árvore esquerda do filho á esquerda do nó desbalanceado, devemos realizar uma rotação simples para a direita, fazendo com que o **nó 5** passe a ser pai do **nó 1** que, por sua vez, passa a ser filho à esquerda do **nó 5**. O **nó 2** deixa de ser filho do **nó 5** e passa a ser filho a direita do **nó 1**. Caso o **nó 1** tenha alguma sub árvore à direita, a raiz dessa sub árvore passa a ser filho à esquerda do **nó 2** (nó 2 adota o filho órfão do nó 1).

Unidade 5

Árvores AVL

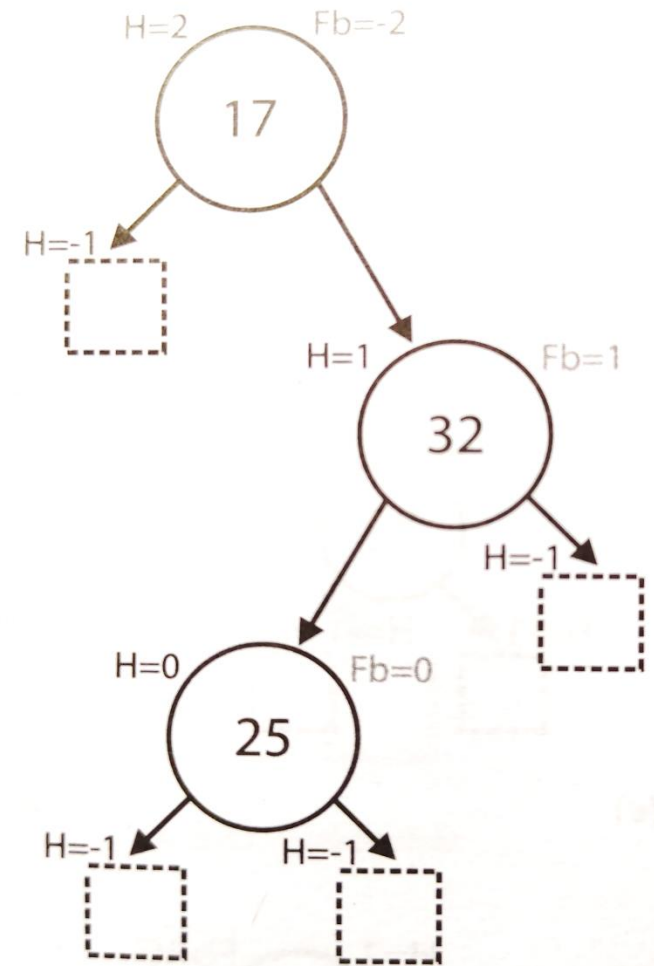
Veja o resultado da rotação:



Unidade 5

Árvores AVL

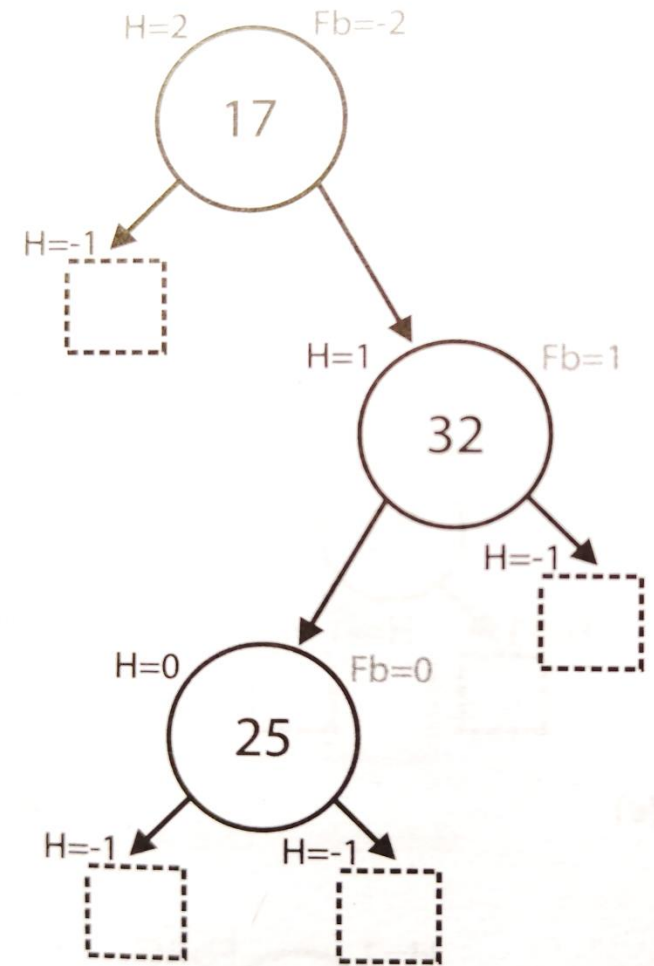
Exemplo: Considere um árvore, na qual os nós 17,32 e 25 foram inseridos na árvore de busca binária, na respectiva ordem. Com isso, temos a raiz 17 se tornando desbalanceada, veja:



Unidade 5

Árvores AVL

A quebra no balanço foi realizada por meio de uma inserção na sub árvore esquerda do filho a direita em relação ao nó desbalanceado, ou seja, o número 25 foi inserido à esquerda o nó 32 que, por sua vez, está a direita do nó 17.



Unidade 5

Árvores AVL

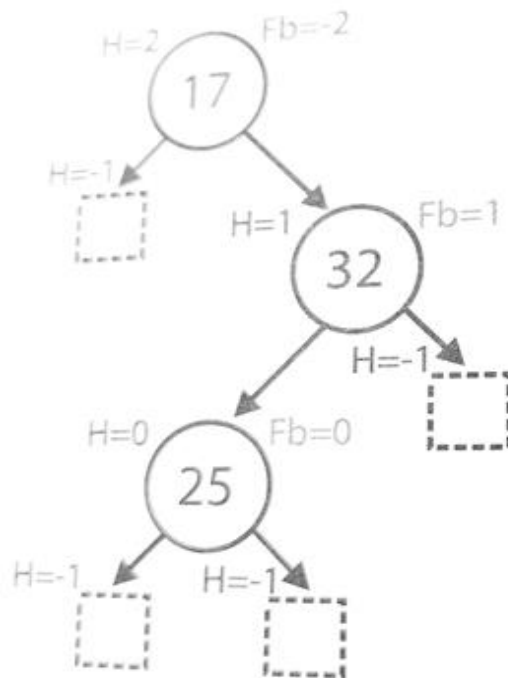
Em uma situação como essas, uma única rotação simples em torno do nó desbalanceado não resolve o problema. Isso ocorre pois, ao tentar realizar uma única rotação à esquerda, continuaríamos tendo o nó raiz, 17, desbalanceado.

Nessas situações é preciso empregar a **rotação dupla direita-esquerda**, que é feita em duas etapas.

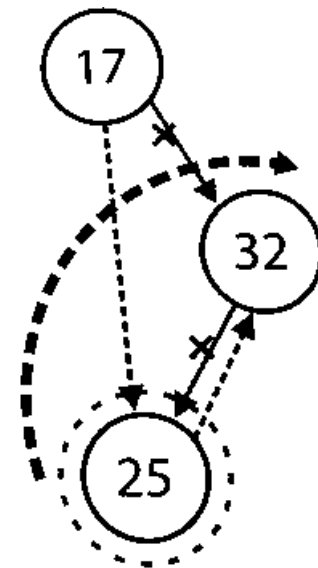
Unidade 5

Árvores AVL

Nesse caso: primeiramente rotaciona-se à direita a sub árvore com a raiz em 25, em direção ao nó 32.



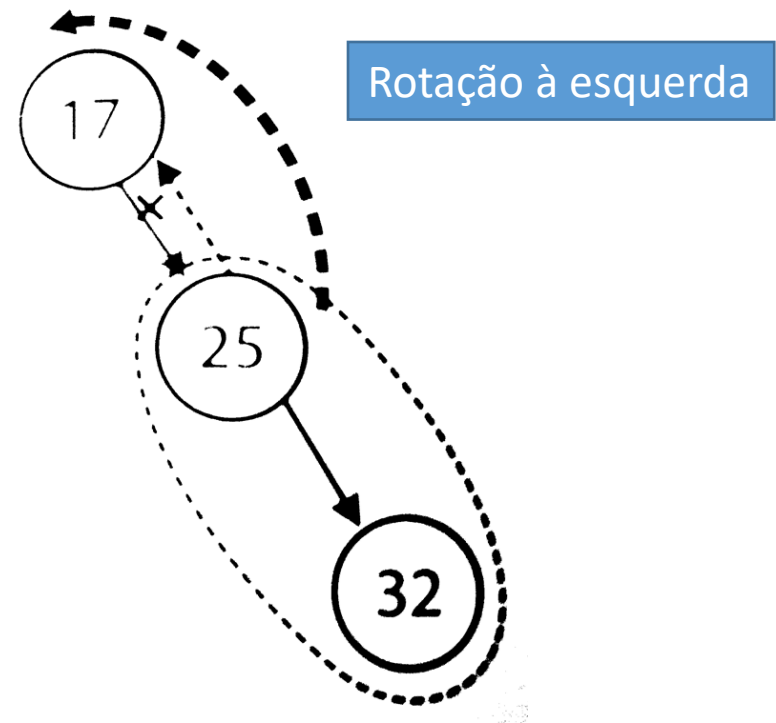
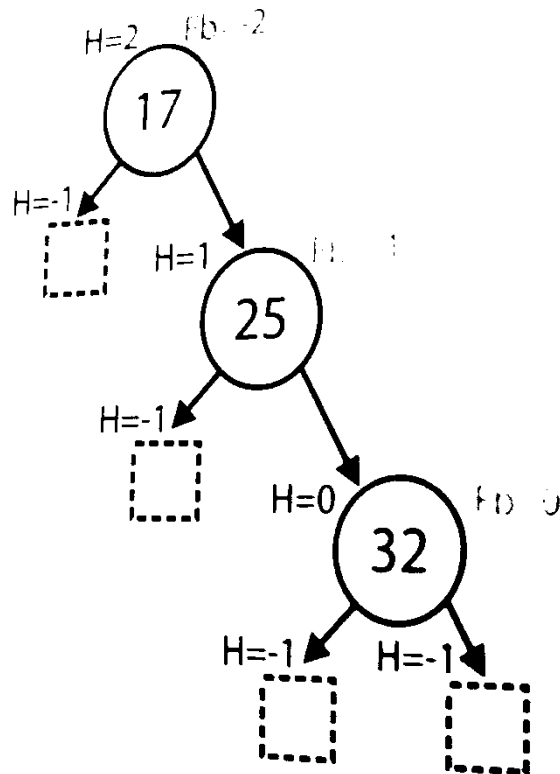
Rotação à direita



Unidade 5

Árvores AVL

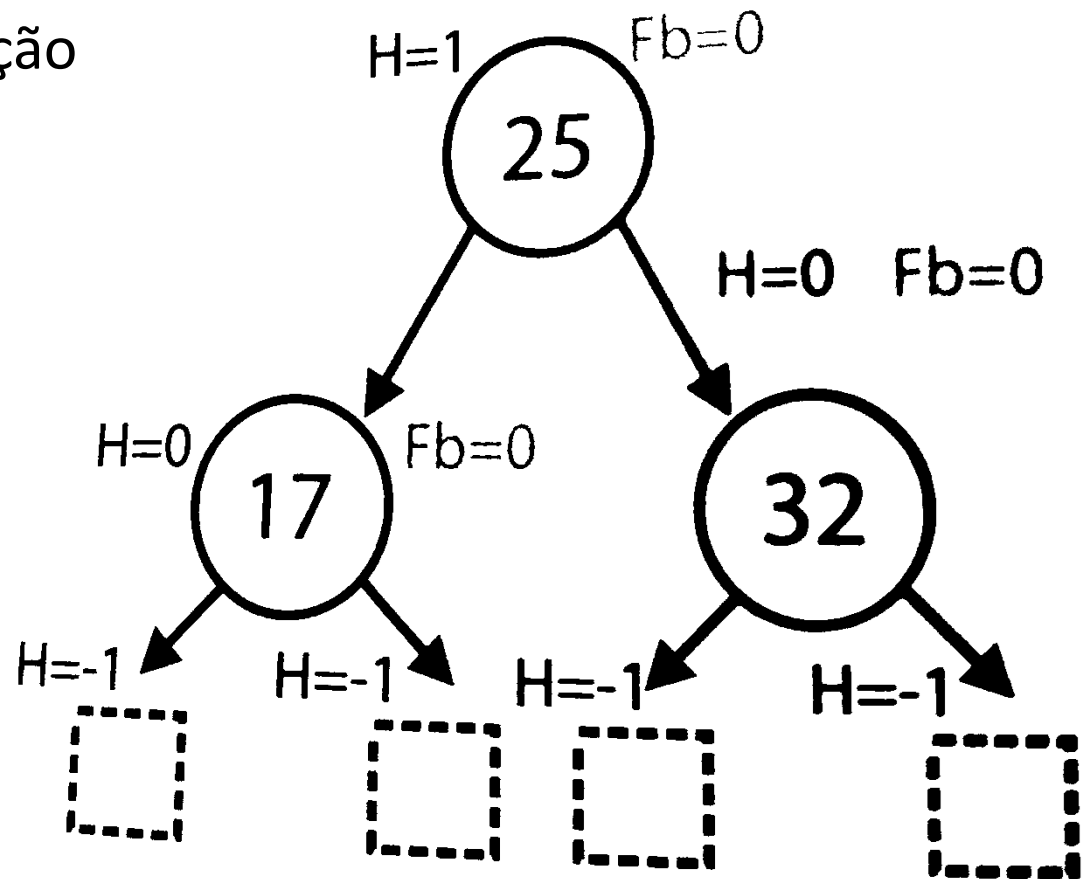
Num segundo momento, rotacionamos à esquerda a sub-árvore com raiz 25, em direção ao nó 17.



Unidade 5

Árvores AVL

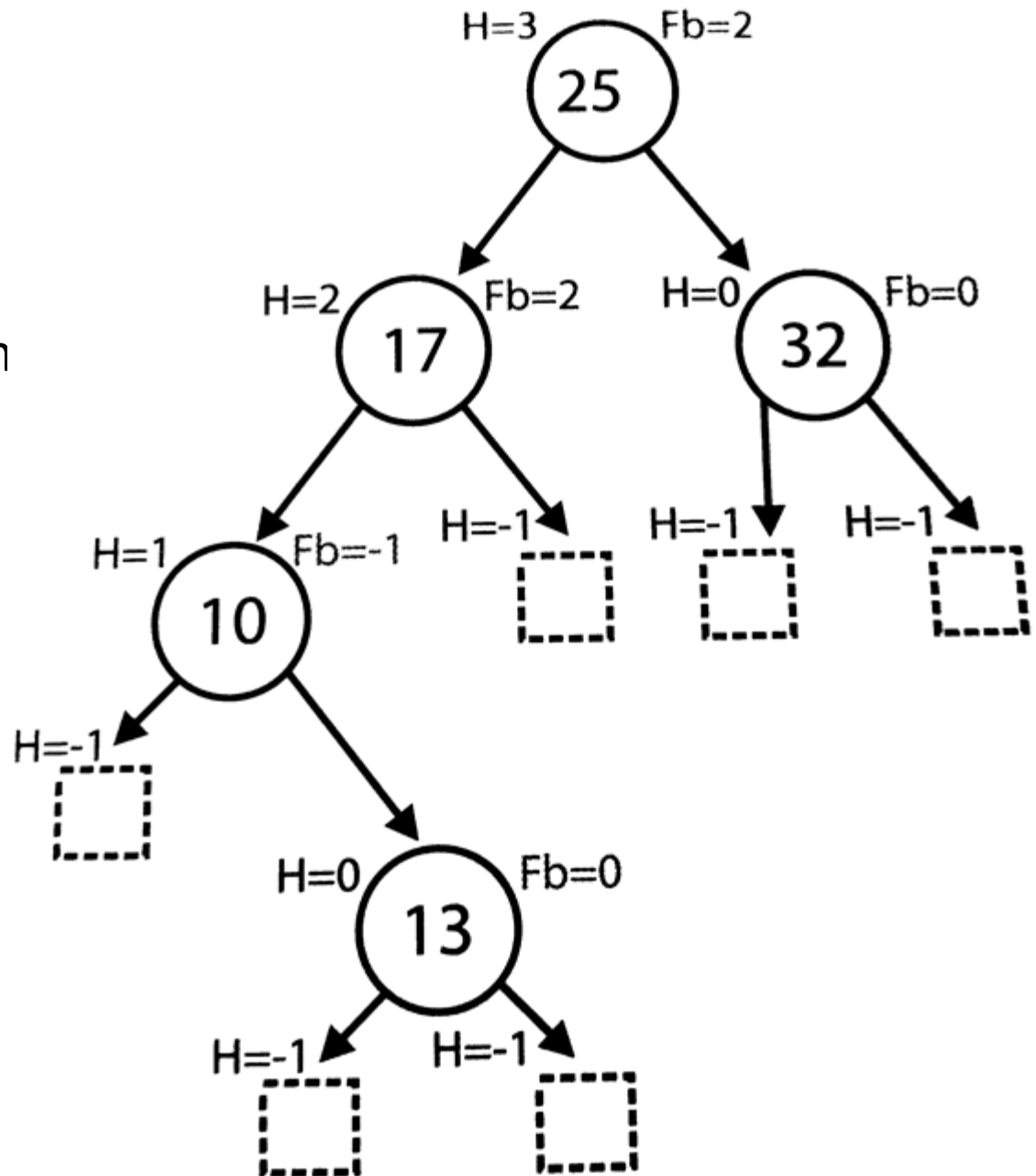
Resultado com a rotação
dupla:



Unidade 5

Árvores AVL

Consideramos agora um inserção de dois nós 10 e 13, um em seguida do outro veja:



Unidade 5

Árvores AVL

Após a inserção do 10 à esquerda do 17, e do 13 à direita do 10, temos uma situação na qual os fatores de balanceamento dos nós 17 e 25 quebram a regra da árvore AVL.

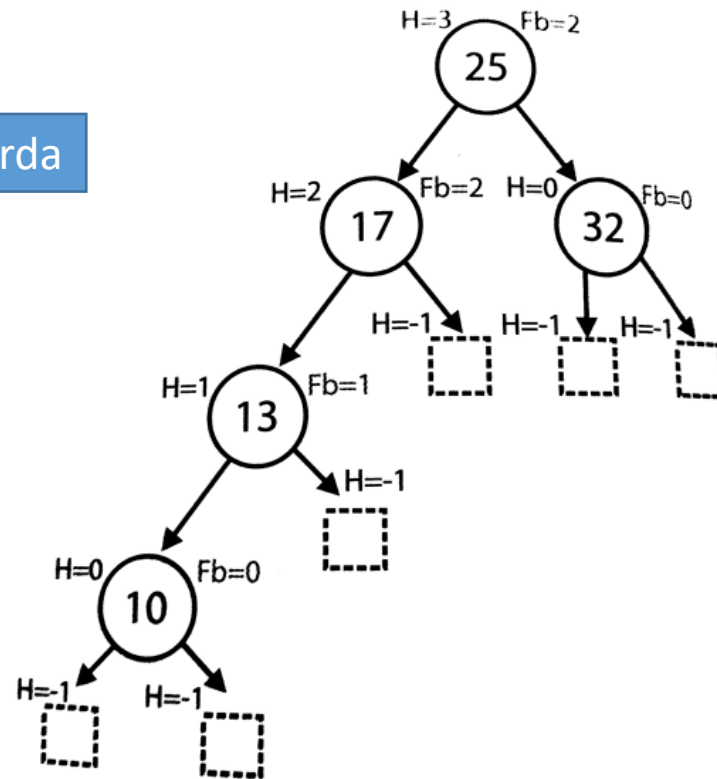
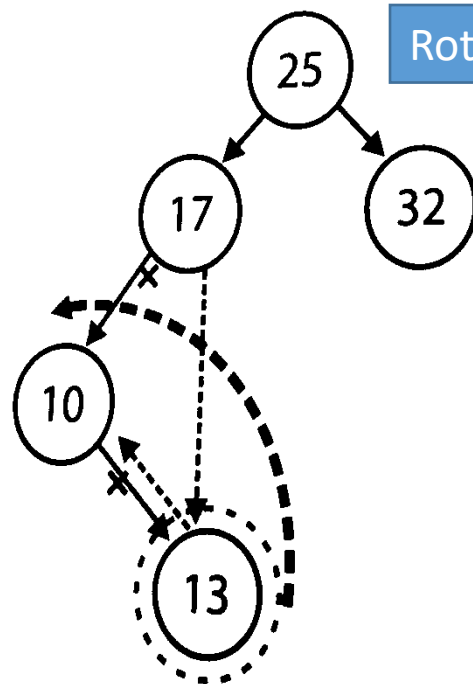
Nesse caso, temos uma inserção na sub árvore direita do filho esquerdo em relação ao nó desbalanceado.

Assim, precisamos nesse caso, realizar rotação dupla esquerda direita , em duas etapas.

Unidade 5

Árvores AVL

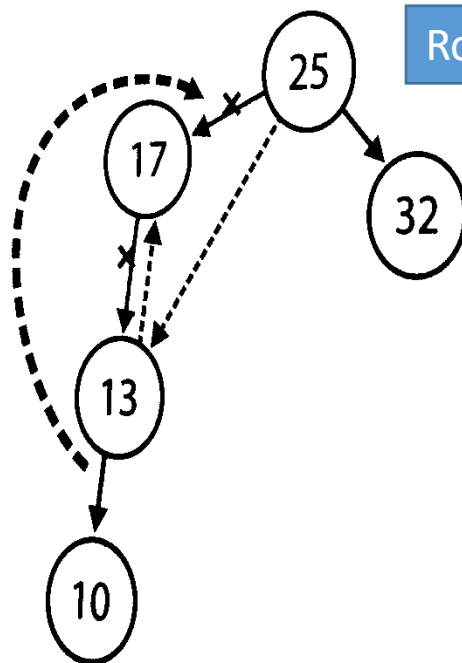
Primeiramente fazemos com que o nó 10 se torne o filho à esquerda do nó 13, veja:



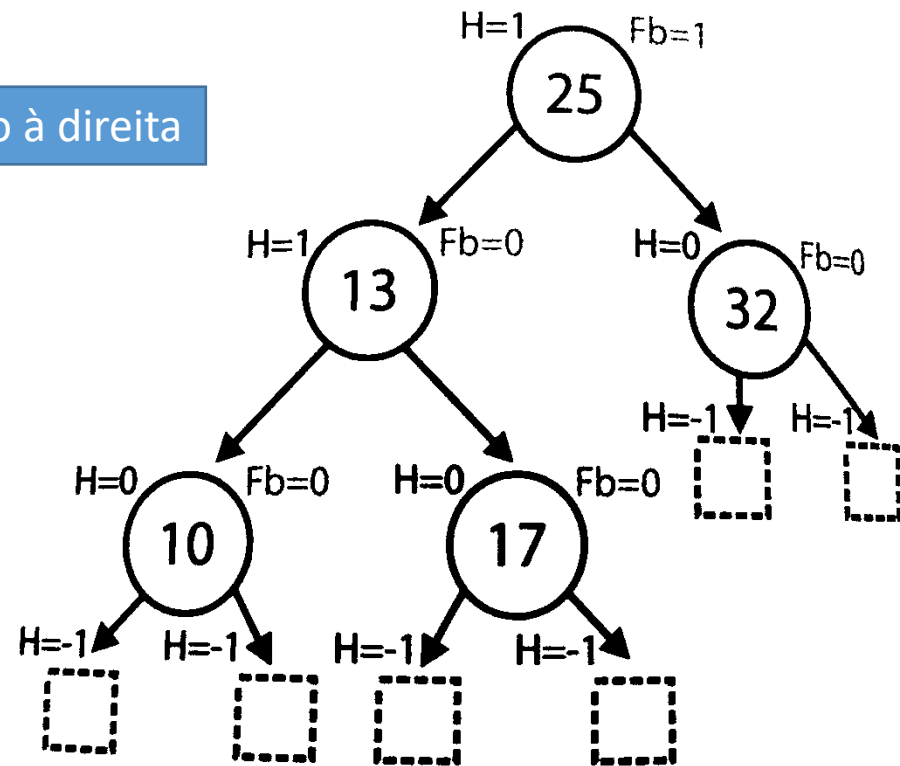
Unidade 5

Árvores AVL

Depois, para balancear o nó 17, realizamos uma rotação a partir do nó 13, veja:



Rotação à direita



Unidade 5

Árvores AVL

Foi apresentado 4 situações de balanceamento AVL:

- Inserção na sub árvore direita do filho à direita: **solução com rotação simples à esquerda.**
- Inserção na sub árvore esquerda do filho à esquerda: **solução com rotação simples à direita.**
- Inserção na sub árvore esquerda do filho à direita: **solução com rotação dupla direita-esquerda.**
- Inserção na sub árvore direita do filho à esquerda: **solução com rotação dupla esquerda-direita.**

Unidade 5

Árvores AVL

Em resumo, uma árvore AVL é uma árvore binária de busca na qual o fator de balanceamento de cada um de seus nós não pode ser maior que 1 ou menor que -1.

Caso o fator de balanceamento de um ou mais nós seja maior ou igual a 2, em valores absolutos, é preciso analisar a ordem das inserções para aplicar a rotação mais adequada para corrigir o balanceamento de cada um dos nós.

Dessa forma a busca em uma árvore AVL se torna mais eficiente.

Unidade 5



Referências:

SZWARCFITER, Jayme Luiz; MARKENZON, Lilian. Estruturas de dados e seus algoritmos. 2ed. Rio de Janeiro: LTC, 1994. 320p.

TENENBAUM, Aaron M.; LANGSAM, Yedidiah; AUGENSTEIN, Moshé J.. Estruturas de dados usando C. São Paulo: Makron Books, 1995. 884p.

VELOSO, Paulo et al.. Estruturas de dados. Rio de Janeiro: Campus, 2001. 228p

Atividades - Unidade 5

