

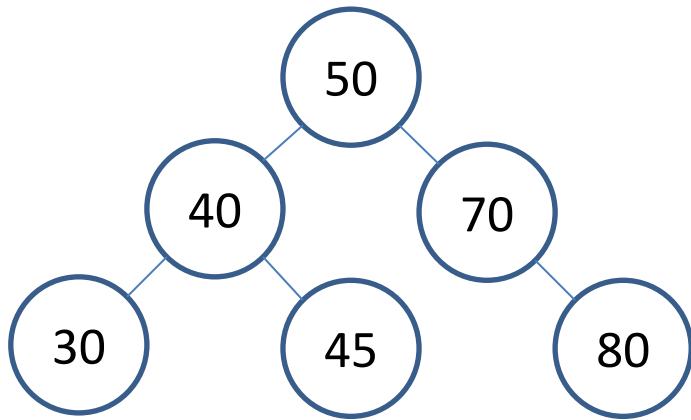
# Árvores AVL

Prof. Martín Vigil

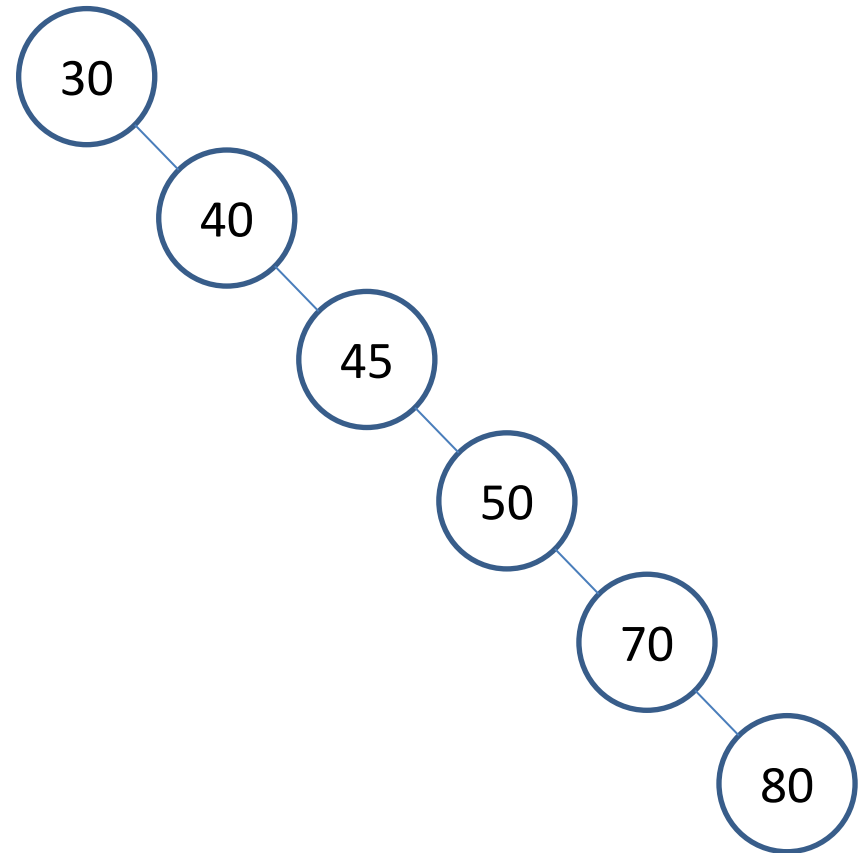
Adaptado de Prof. Ricardo Luís Lachi

<https://www.comp.uems.br/~ricardo/>

# Custo de Árvore Binárias

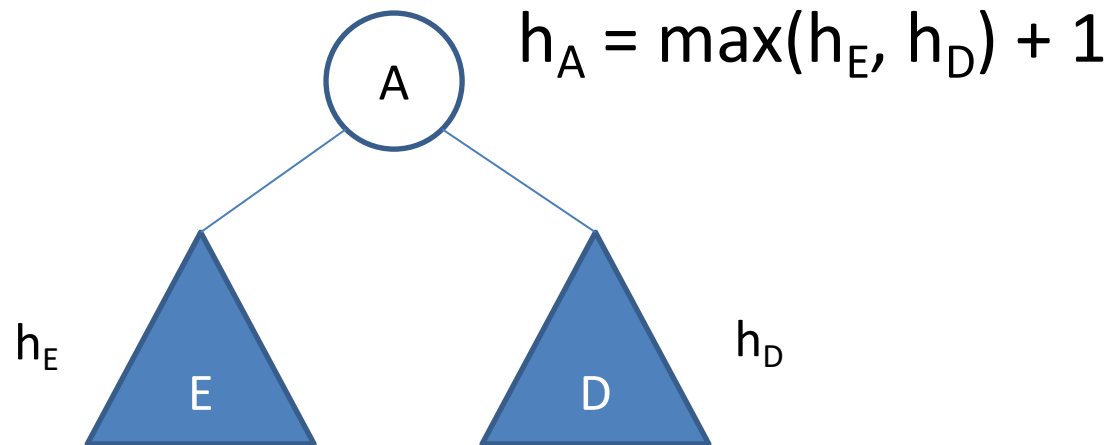


$O(\log n)$

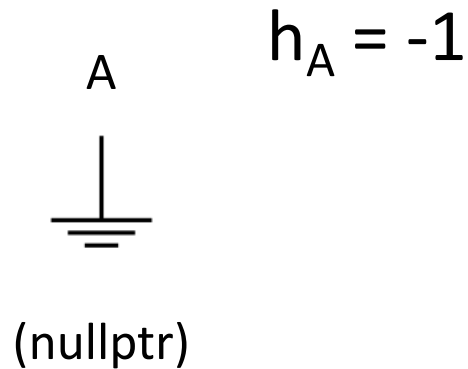


$O(n)$

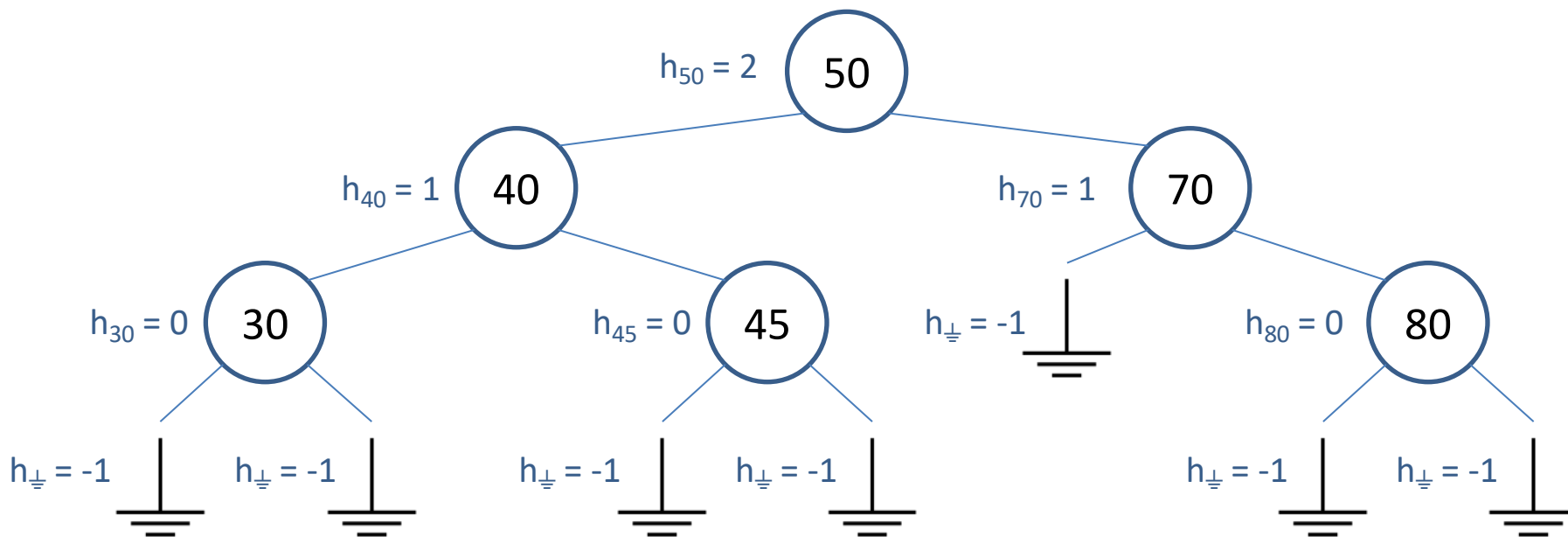
# Altura $h$ da árvore (definição recursiva)



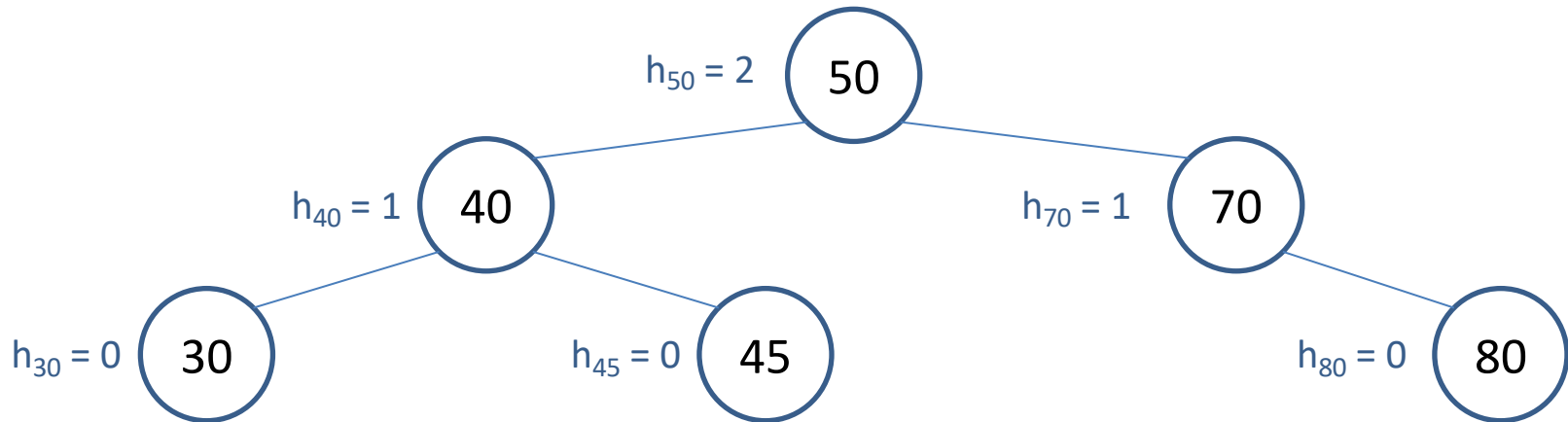
# Altura $h$ da árvore (definição recursiva)



# Exemplo de altura da árvore



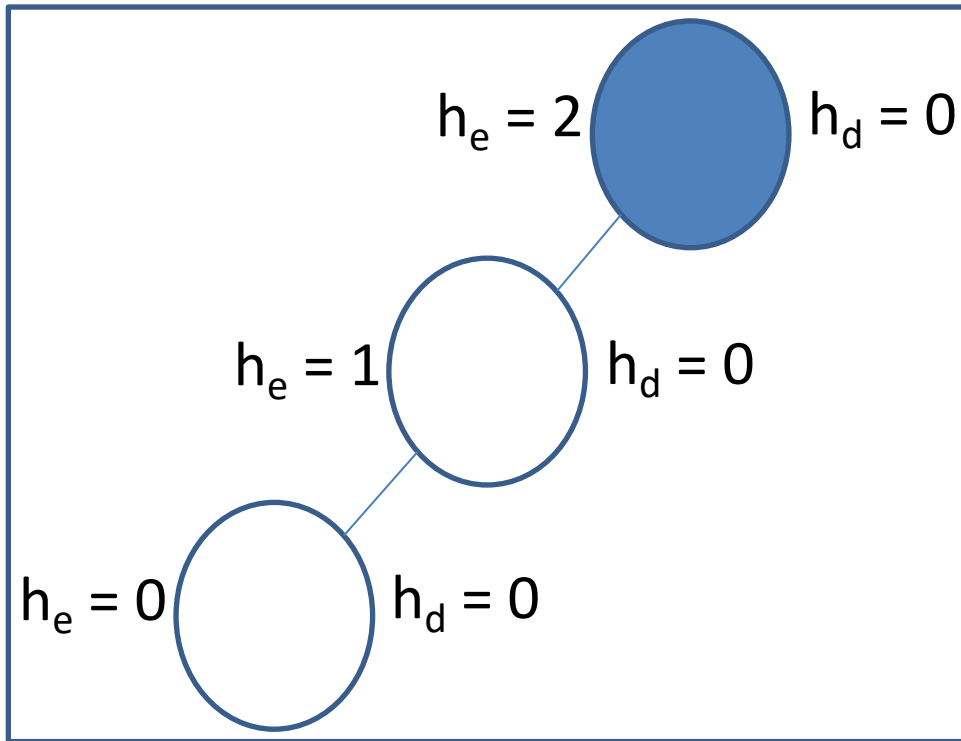
# Exemplo de altura da árvore



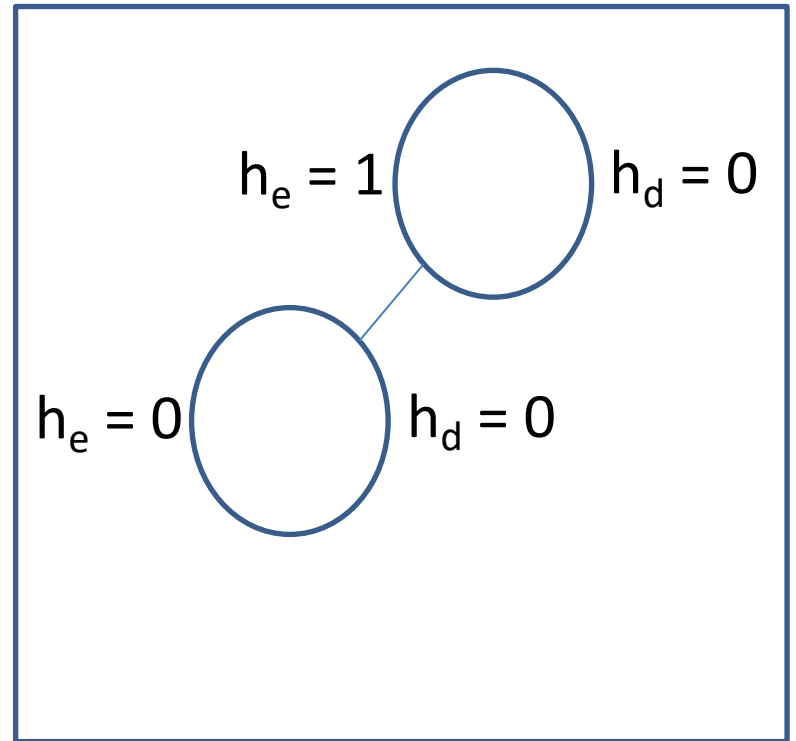
# Árvores AVL

- Propostas pelos soviéticos Georgy Adelson-Velsky e Evgenii Landis em 1962
- São árvores binárias
- Para todo nodo da AVL,  $-1 \leq h_E - h_D \leq 1$
- Parcialmente balanceadas
- Custo  $O(\log n)$  para inserir, remover e buscar

# Exemplos



**Não é AVL**



**AVL**



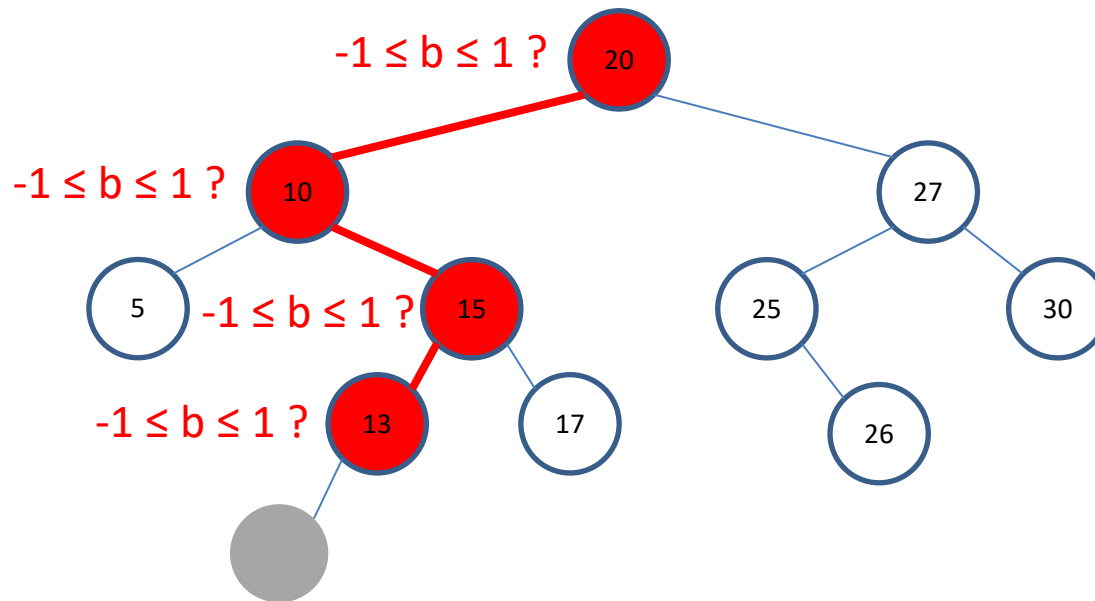
# Árv. AVL *estendem* Árv. Binárias

- Mesmo algoritmo de busca
- Inserção e remoção
  - Função **recursiva** para inserir/remover
  - Rotacionar **antes** do retorno da função se necess.
  - Rotação de acordo com fator de balanceamento b
- Existem quatro tipos de rotações possíveis

# Fator de balanceamento $b$

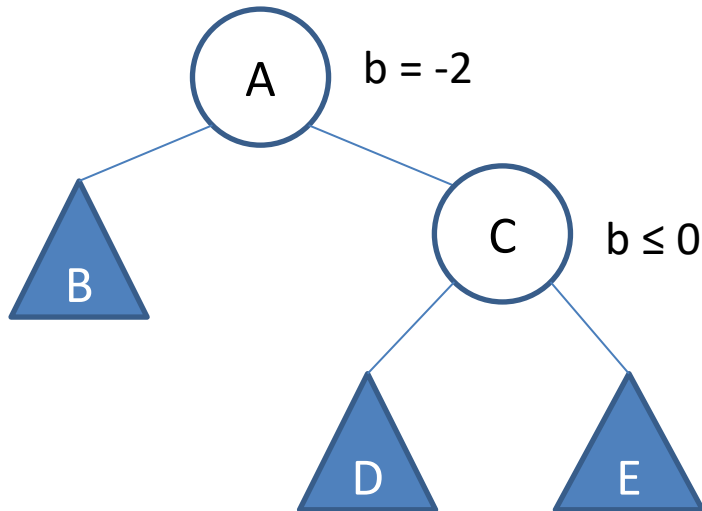
- Fator  $b = h_E - h_D$
- Nodo balanceado sse  $-1 \leq b \leq 1$

# Verificar $b$ no caminho até raiz



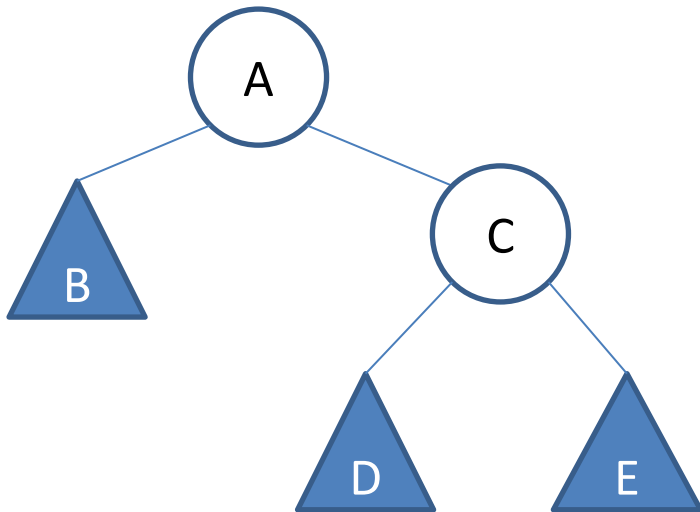
**ROTAÇÃO SIMPLES À ESQUERDA**

# Rotação simples à esquerda

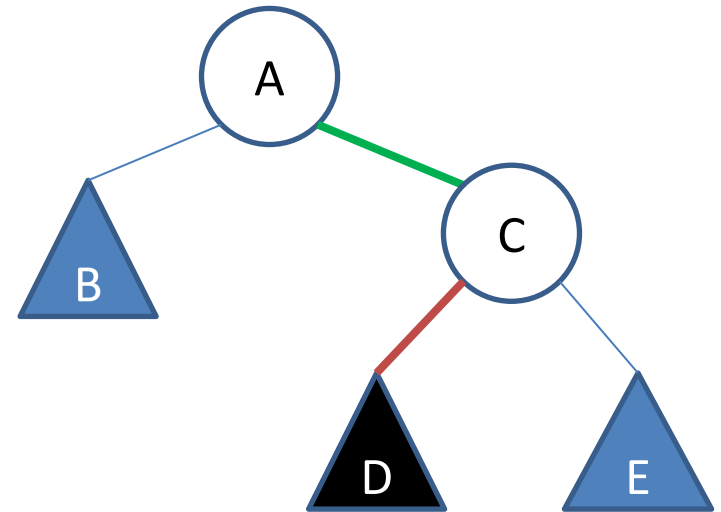


Antes da rotação

# Rotação simples à esquerda

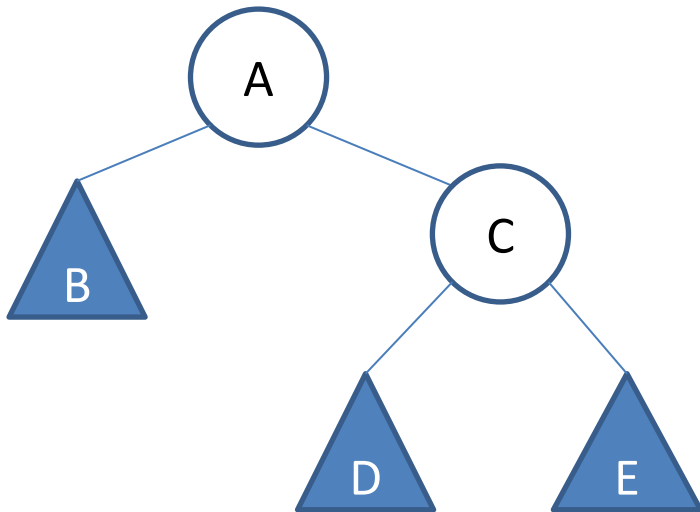


Antes da rotação

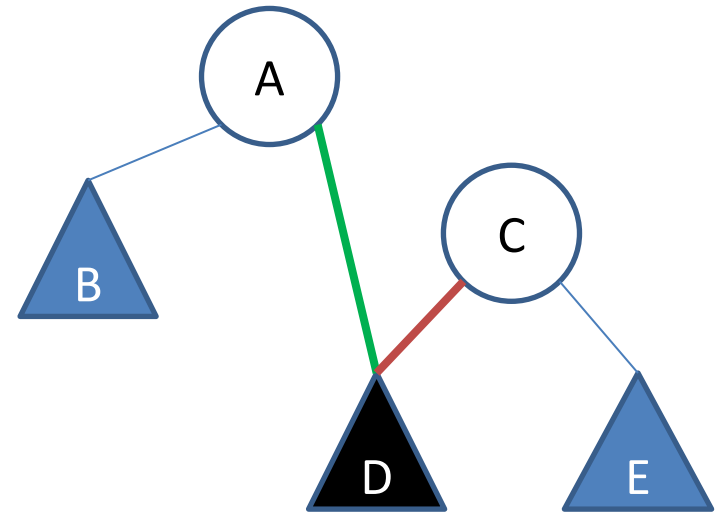


Antes da rotação

# Rotação simples à esquerda (1/2)

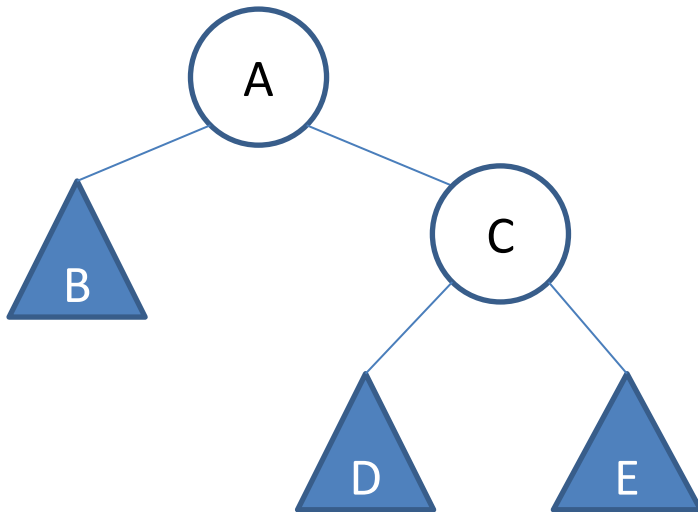


Antes da rotação

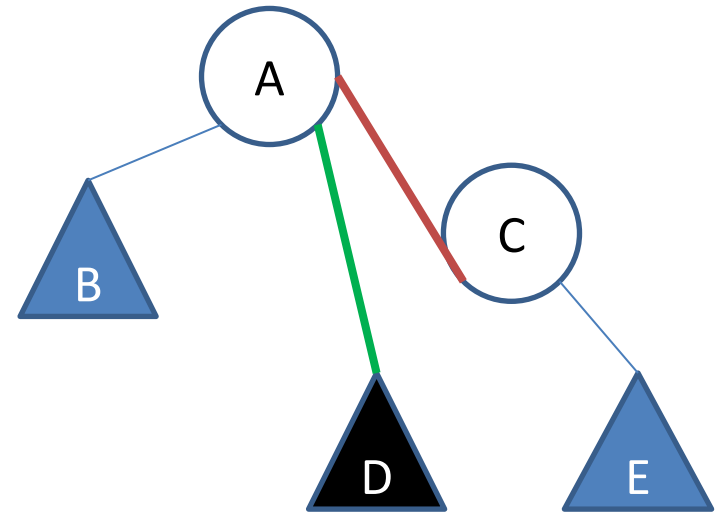


Durante a rotação

# Rotação simples à esquerda (2/2)



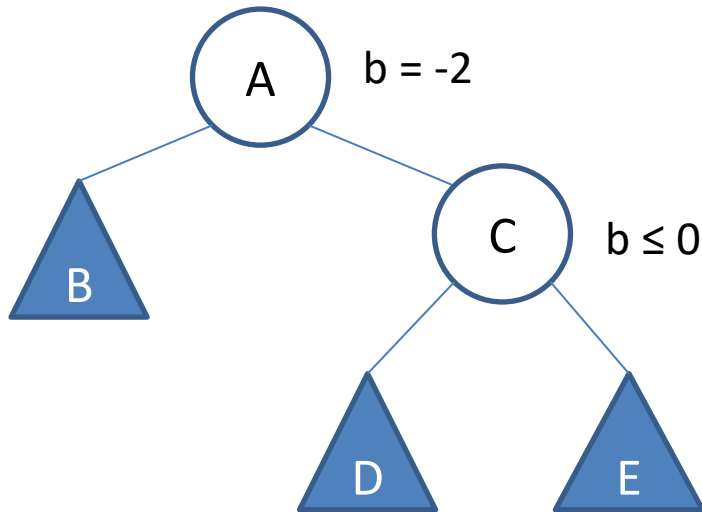
Antes da rotação



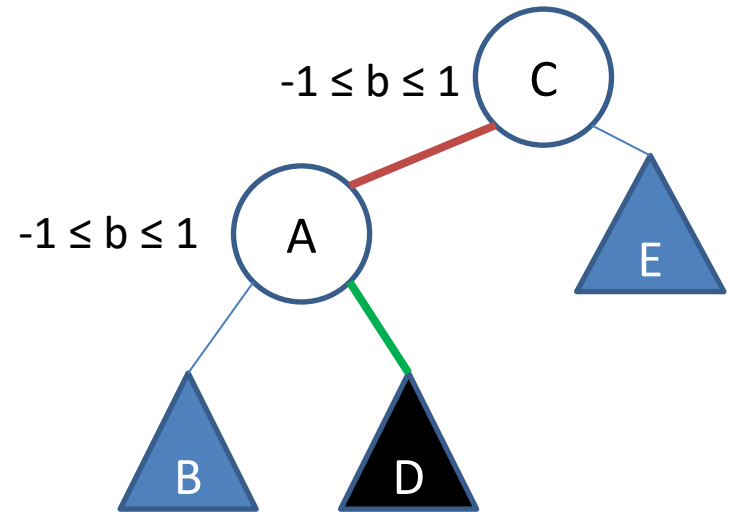
Durante a rotação



# Rotação simples à esquerda (2/2)



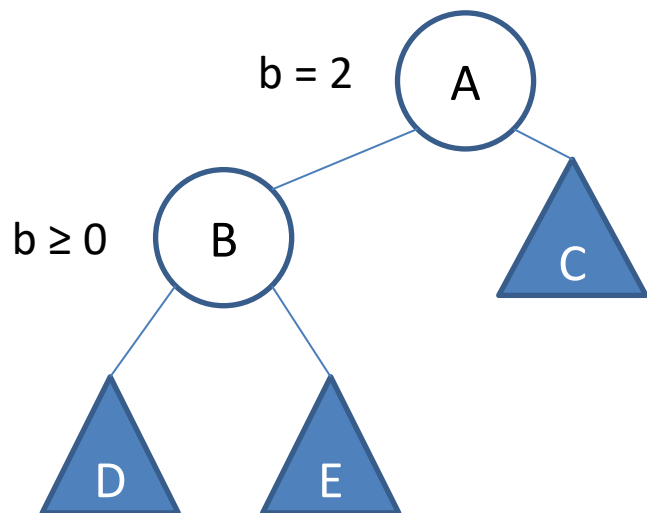
Antes da rotação



Depois da rotação

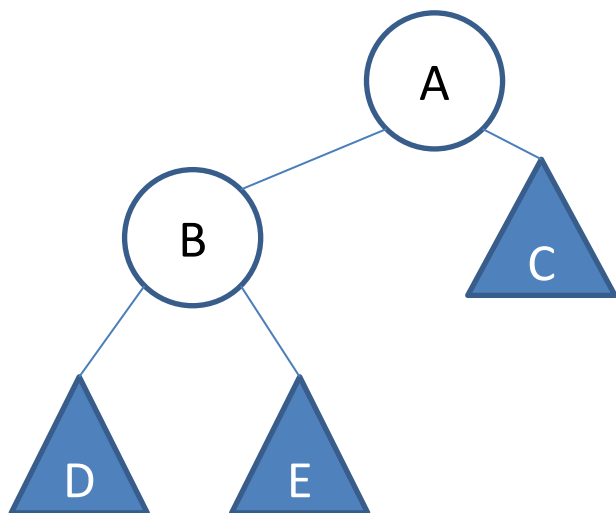
**ROTAÇÃO SIMPLES À DIREITA**

# Rotação simples à direita

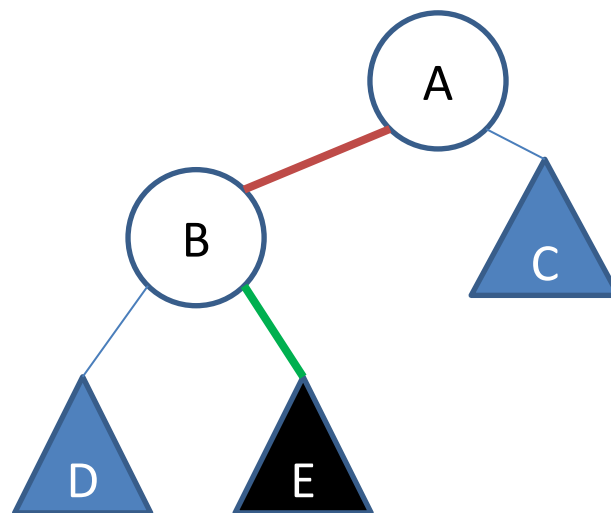


Antes da rotação

# Rotação simples à direita

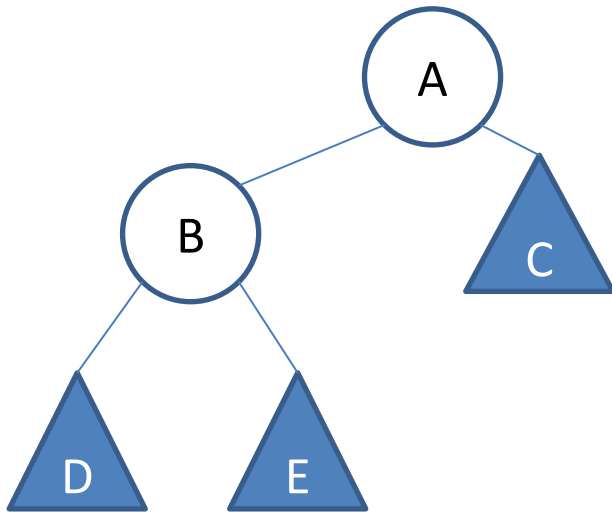


Antes da rotação

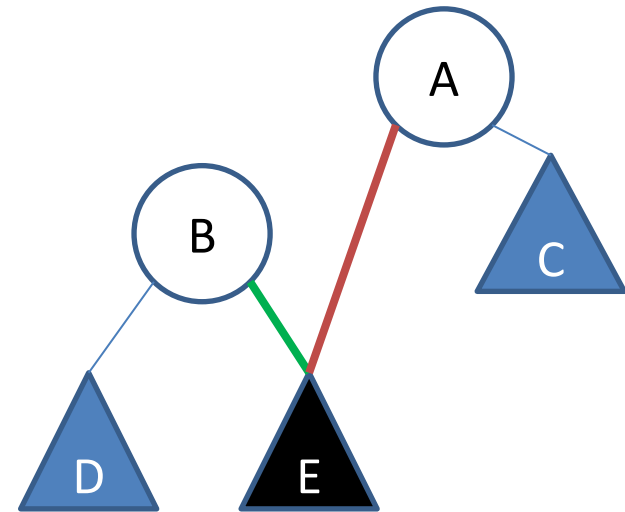


Antes da rotação

# Rotação simples à direita (1/2)

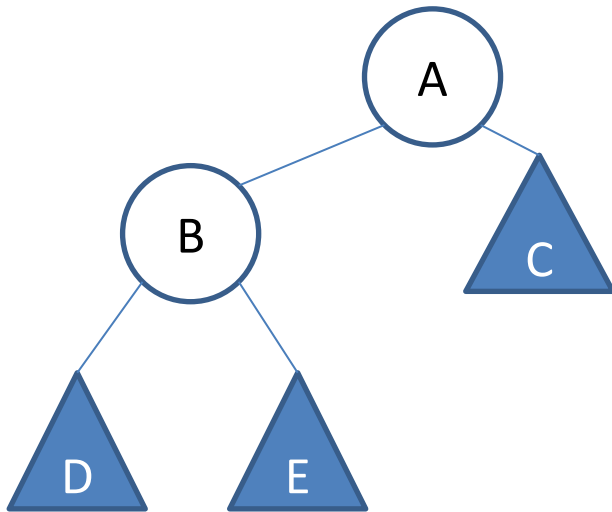


Antes da rotação

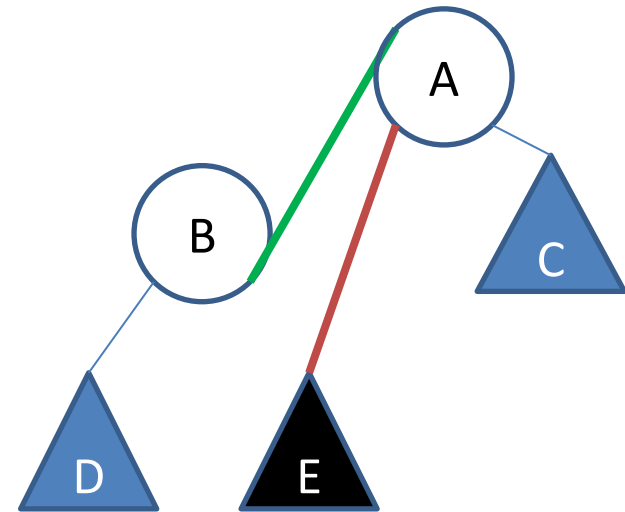


Durante a rotação

# Rotação simples à direita (2/2)

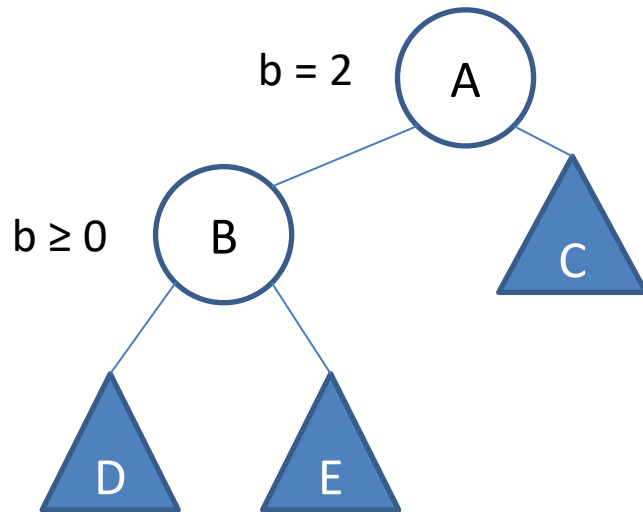


Antes da rotação

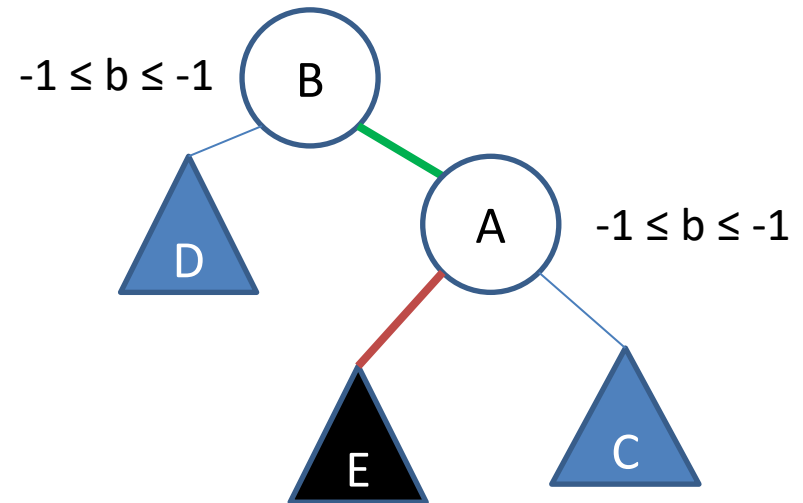


Durante a rotação

# Rotação simples à direita (2/2)



Antes da rotação

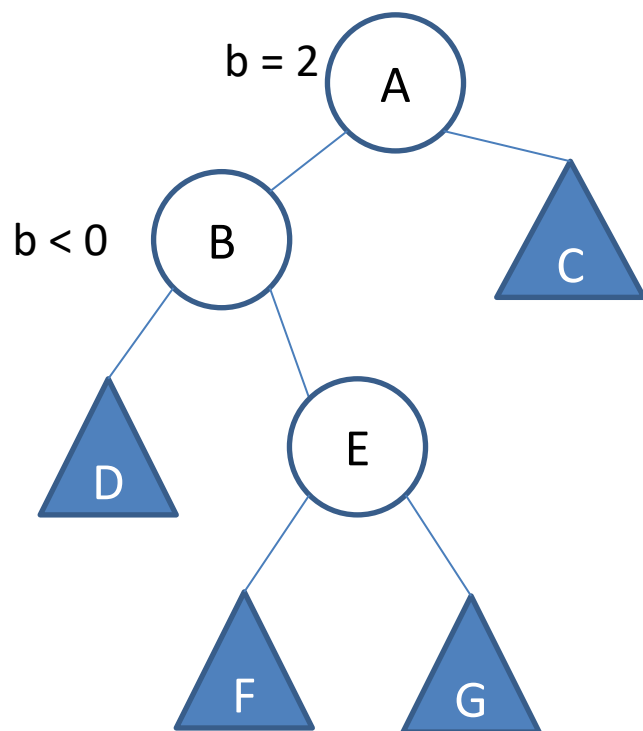


Depois da rotação

**ROTAÇÃO ESQUERDA-DIREITA**

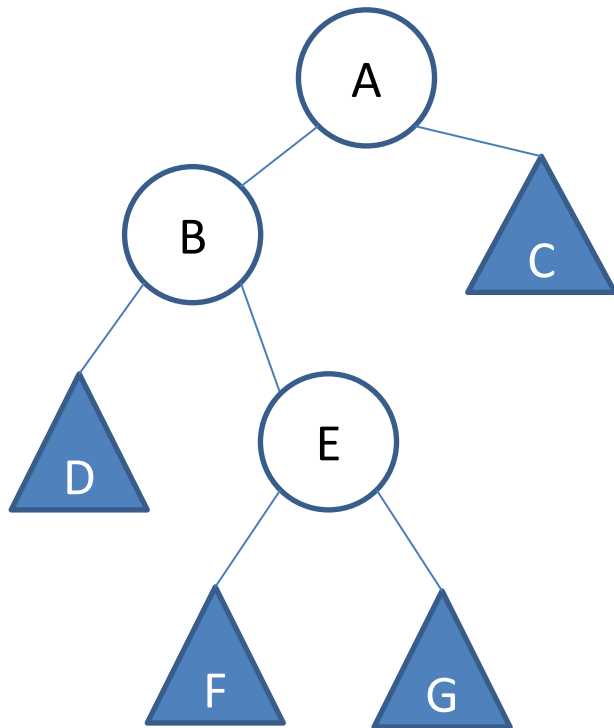


# Rotação esquerda-direita

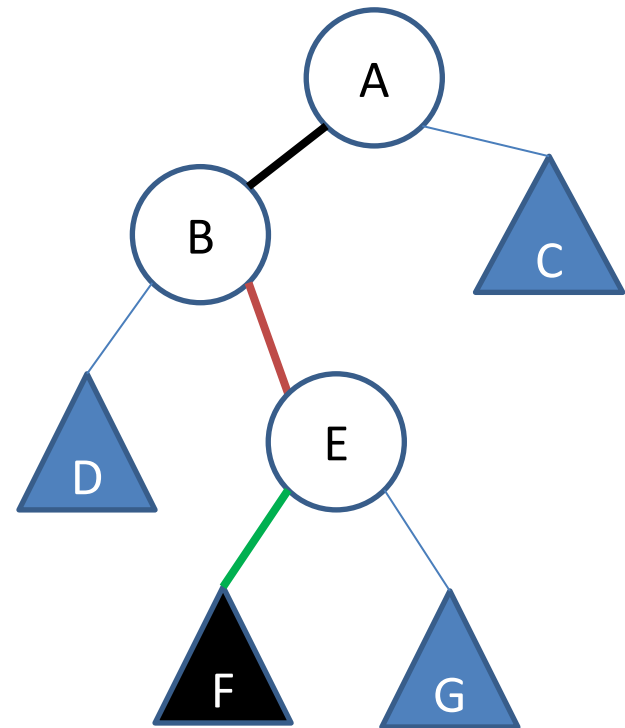


Antes da rotação

# Rotação esquerda-direita

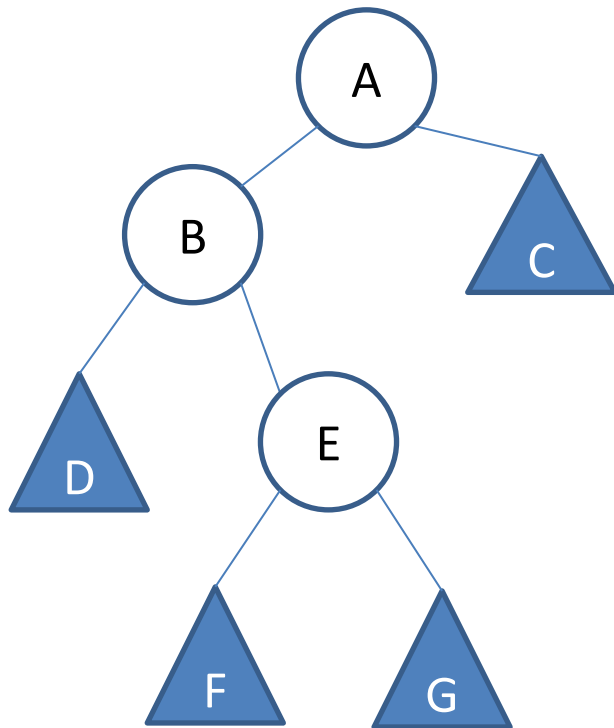


Antes da rotação

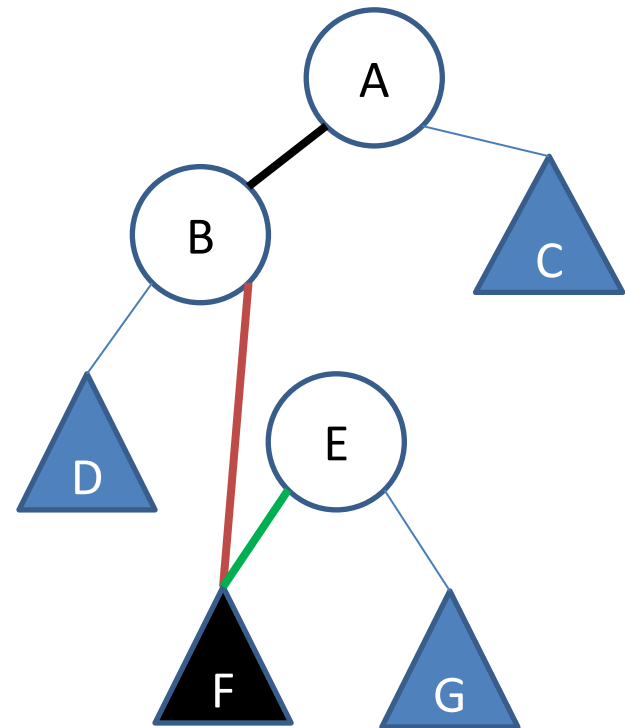


Antes da rotação

# Rotação esquerda-direita (1/5)

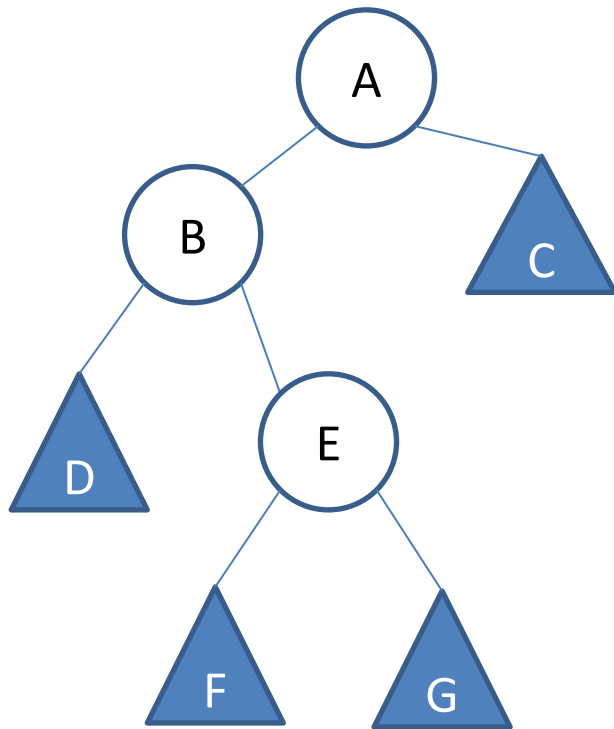


Antes da rotação

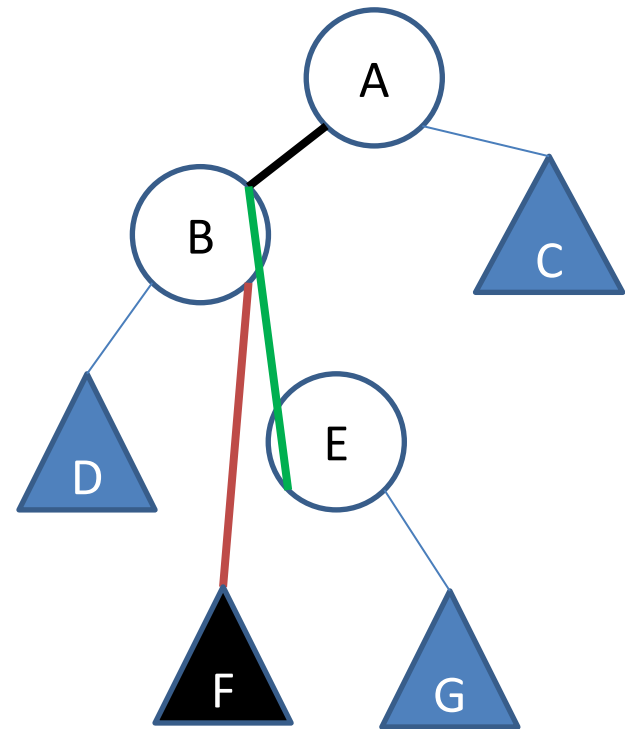


Durante rotação

# Rotação esquerda-direita (2/5)

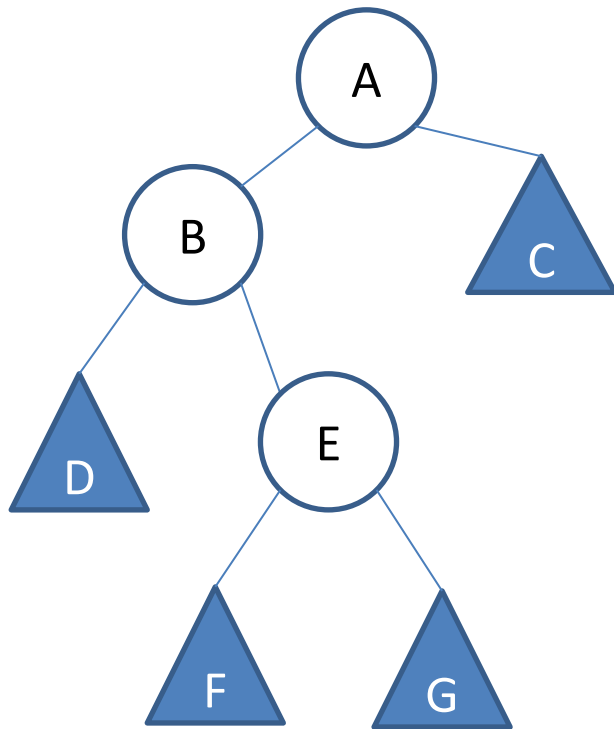


Antes da rotação

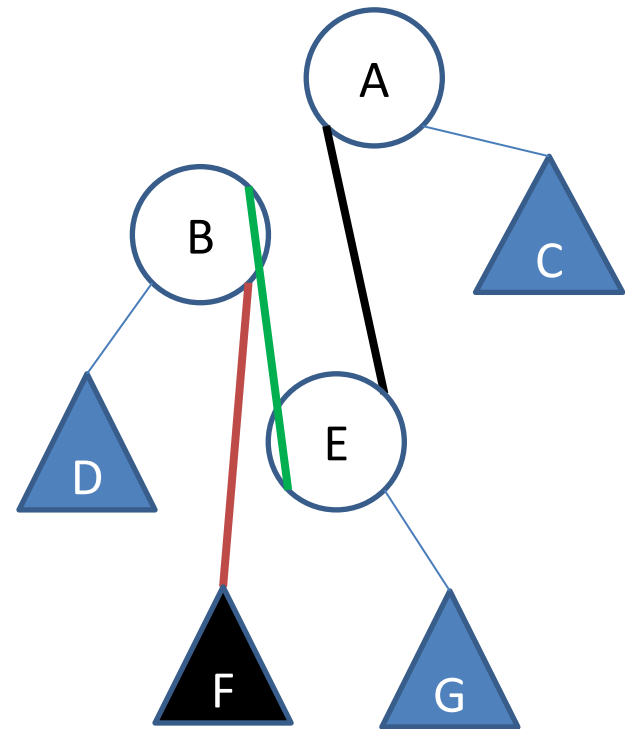


Durante rotação

# Rotação esquerda-direita (3/5)

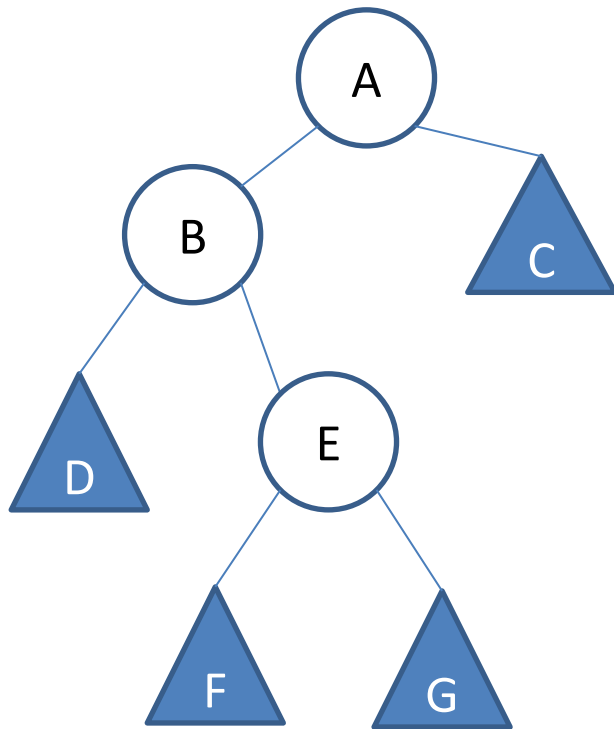


Antes da rotação

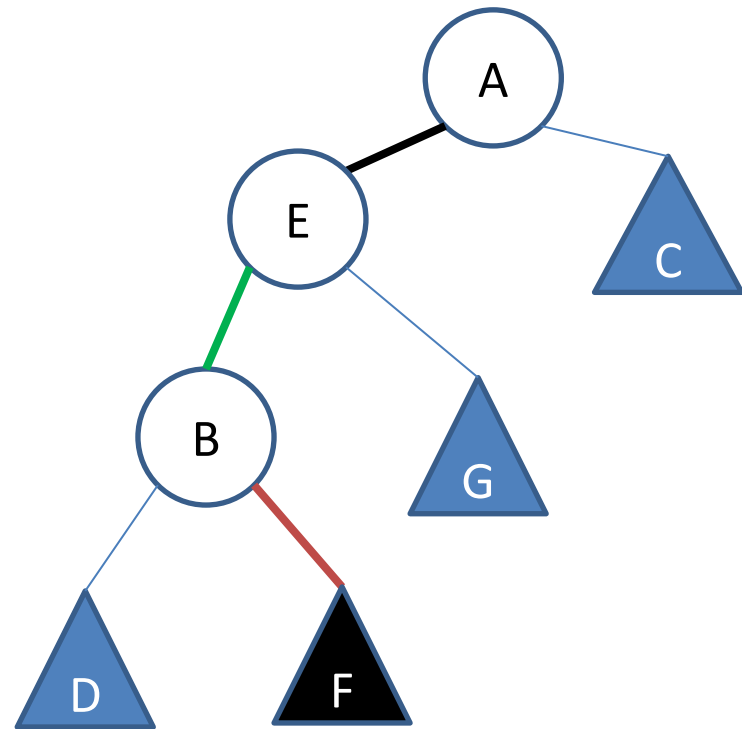


Durante rotação

# Rotação esquerda-direita (3/5)

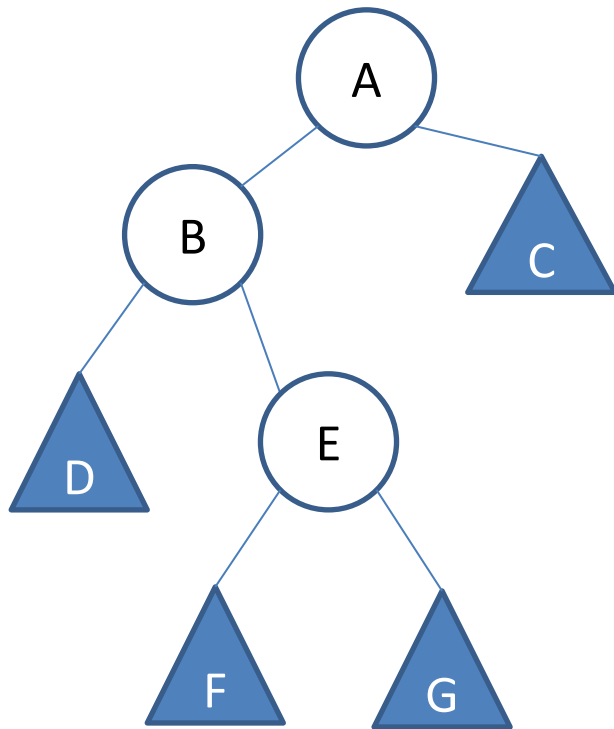


Antes da rotação

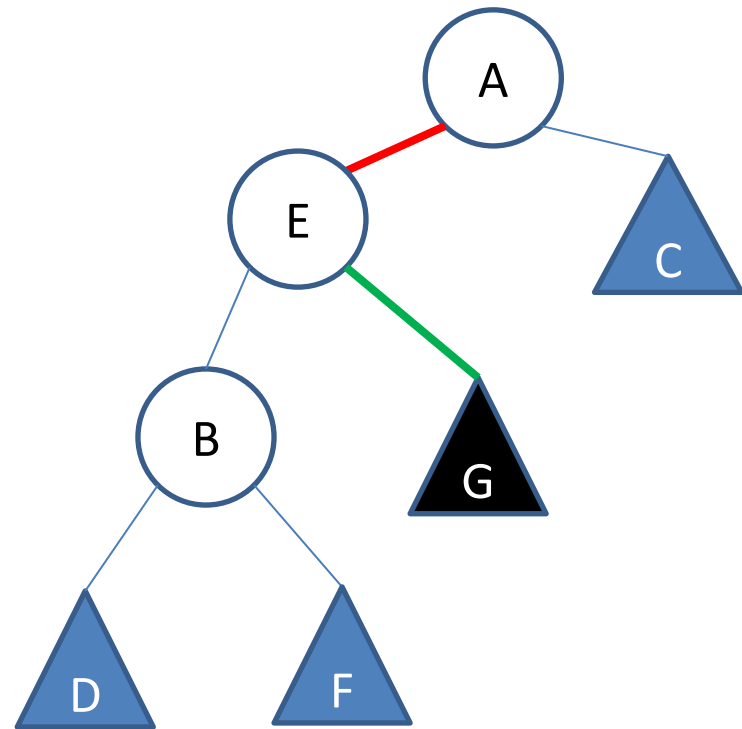


Durante rotação

# Rotação esquerda-direita (3/5)

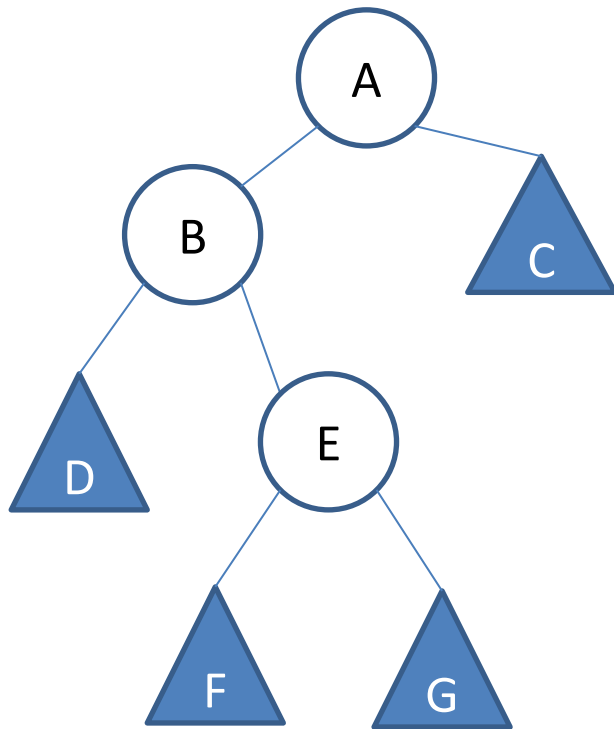


Antes da rotação

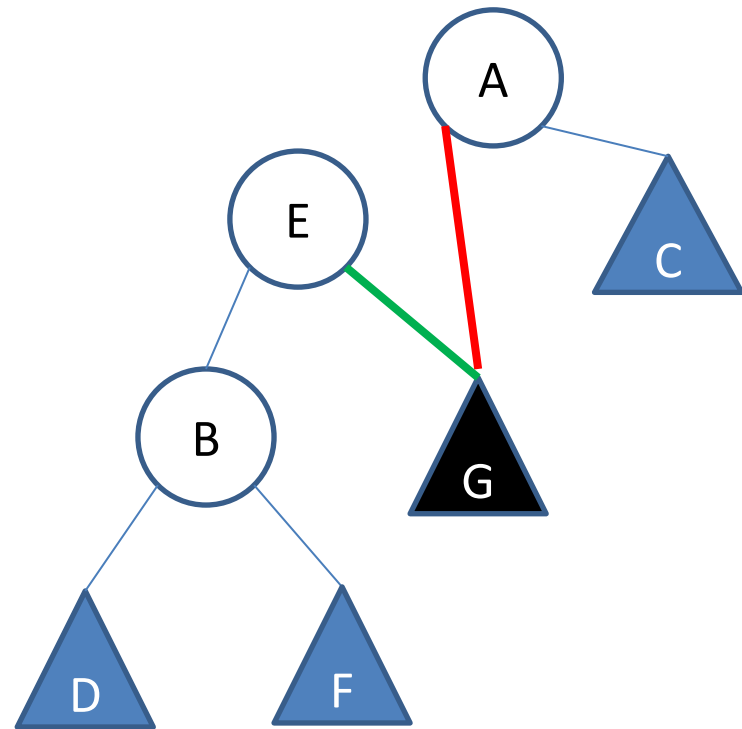


Durante rotação

# Rotação esquerda-direita (4/5)



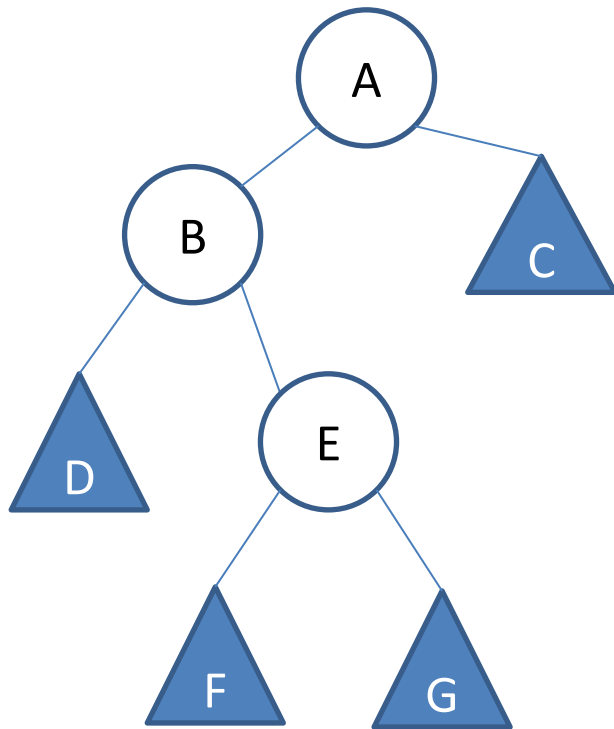
Antes da rotação



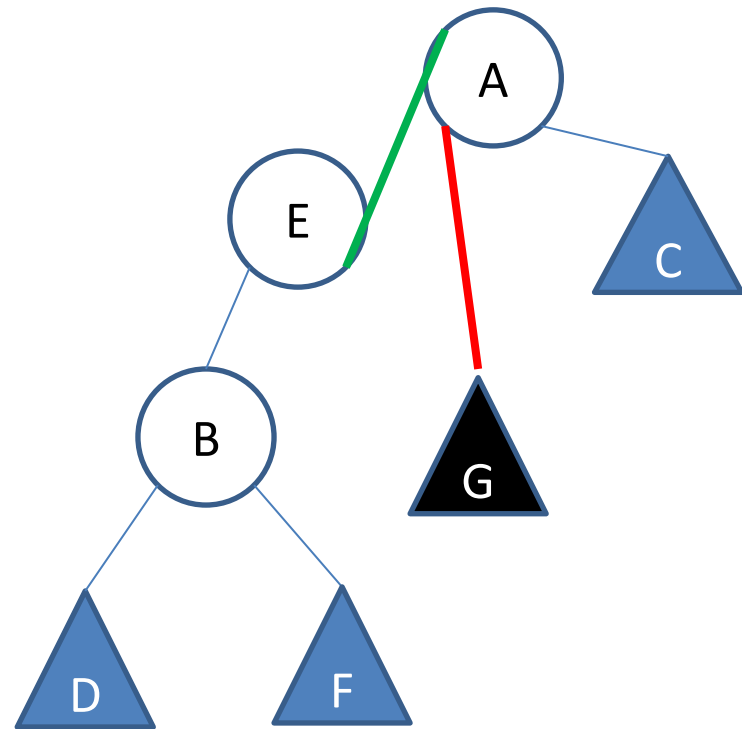
Durante rotação



# Rotação esquerda-direita (5/5)

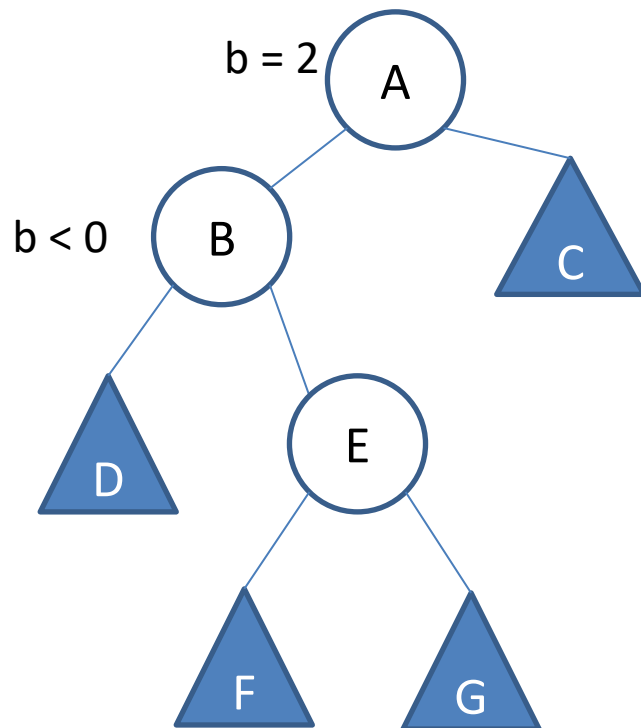


Antes da rotação

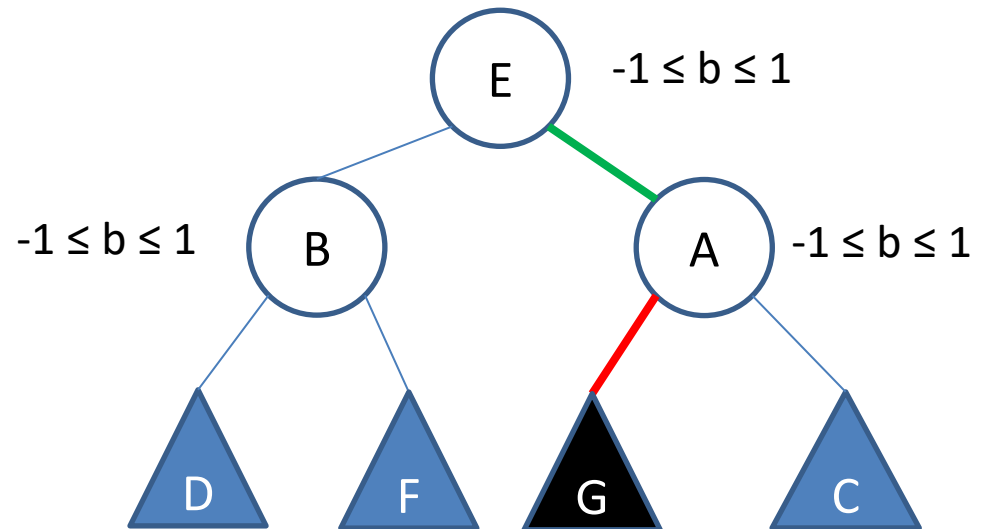


Durante rotação

# Rotação esquerda-direita (5/5)



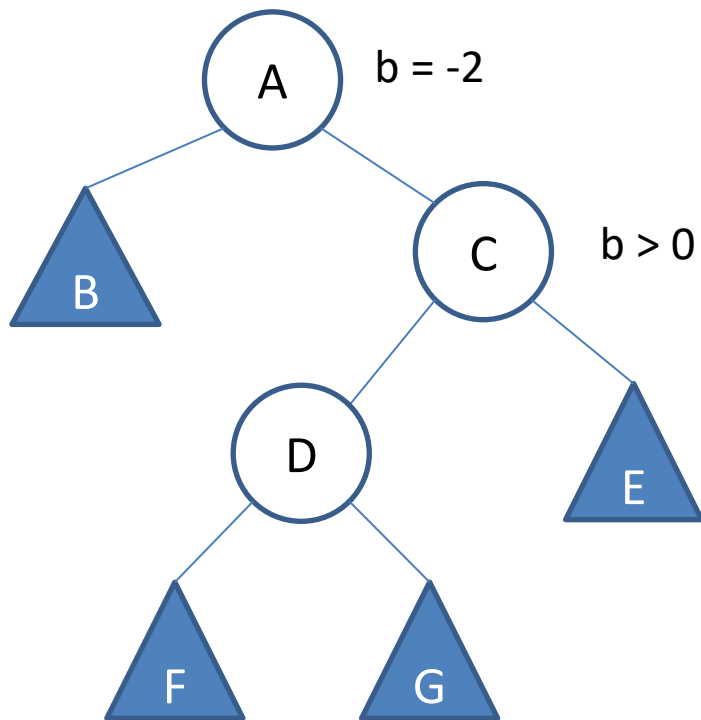
Antes da rotação



Depois da rotação

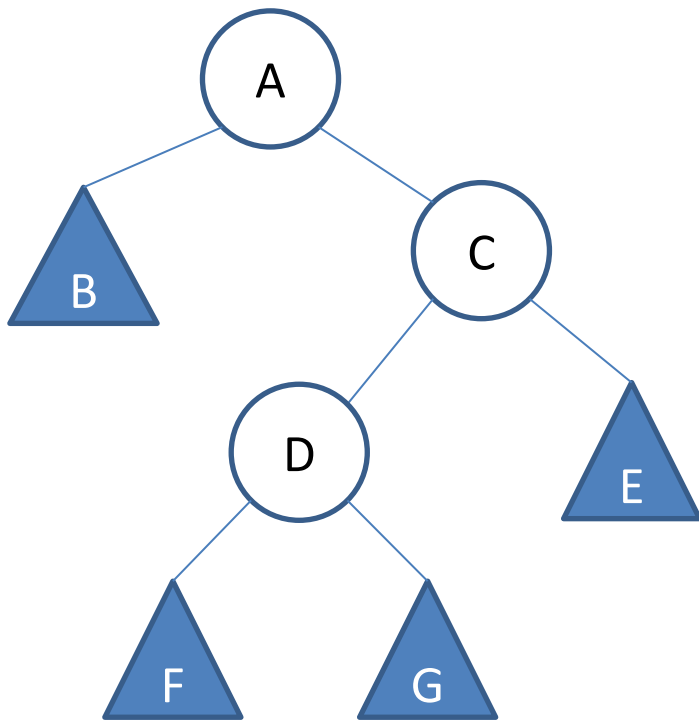
**ROTAÇÃO DIREITA-ESQUERDA**

# Rotação direita-esquerda

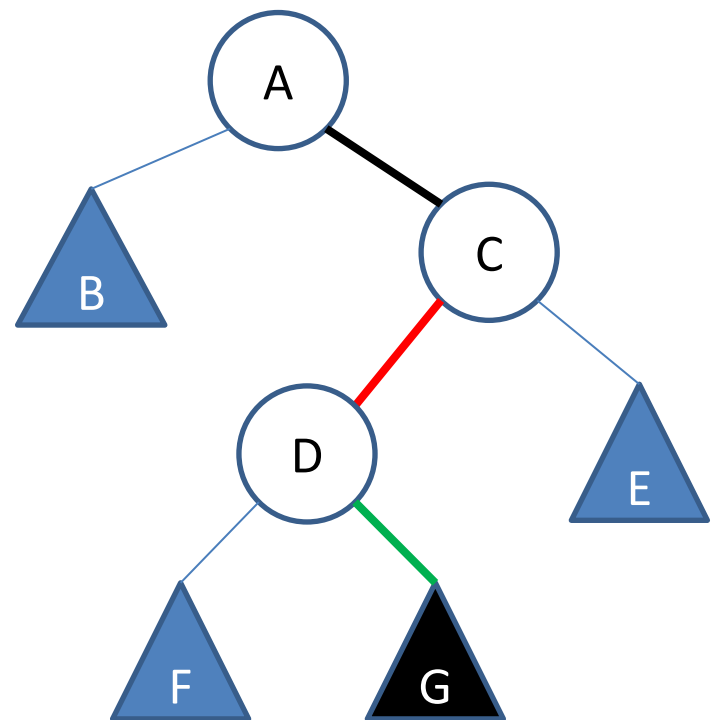


Antes da rotação

# Rotação direita-esquerda

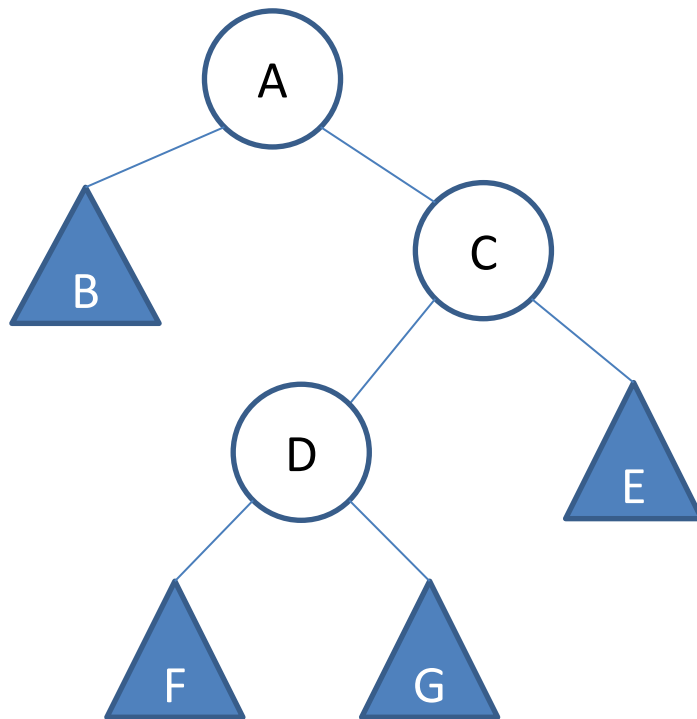


Antes da rotação

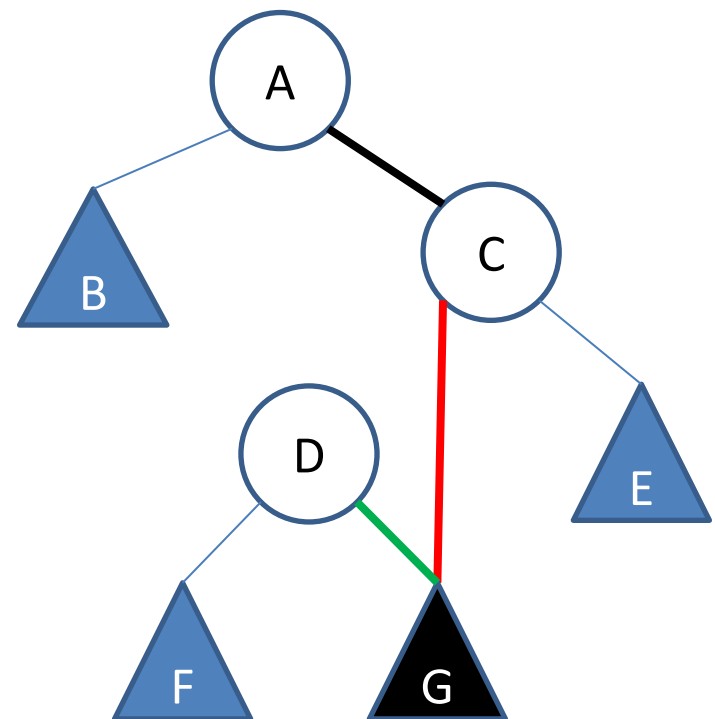


Antes da rotação

# Rotação direita-esquerda (1/5)

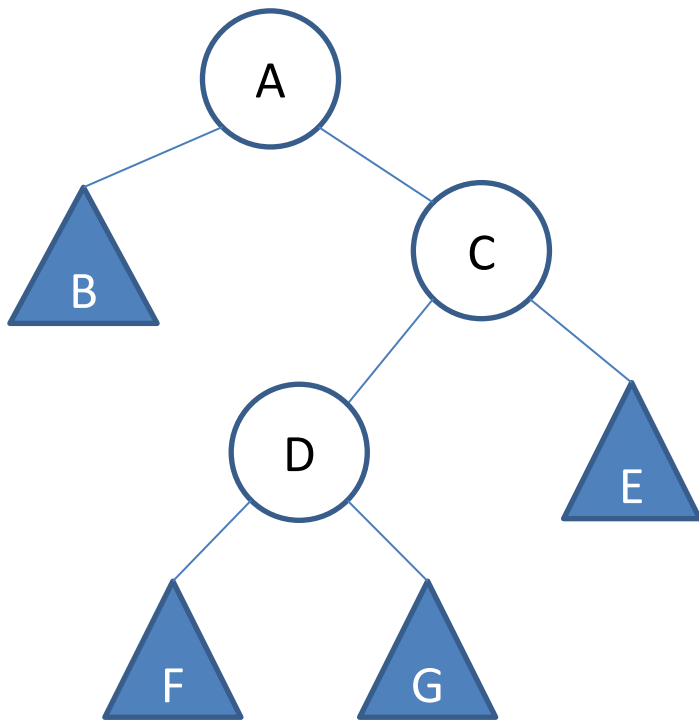


Antes da rotação

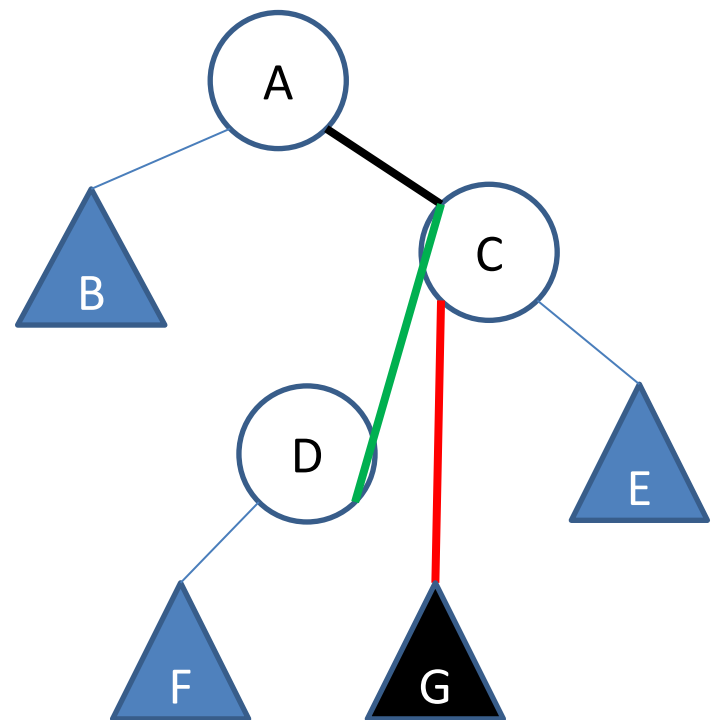


Durante a rotação

# Rotação direita-esquerda (2/5)

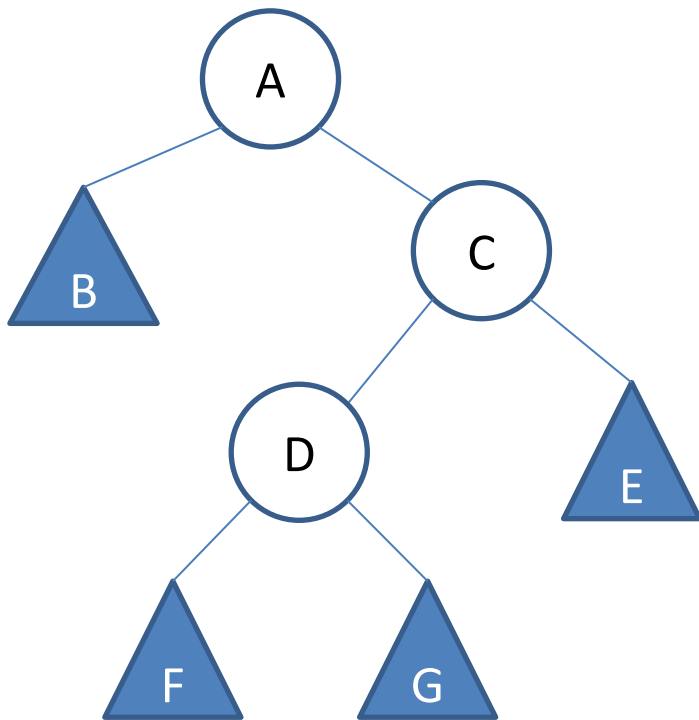


Antes da rotação

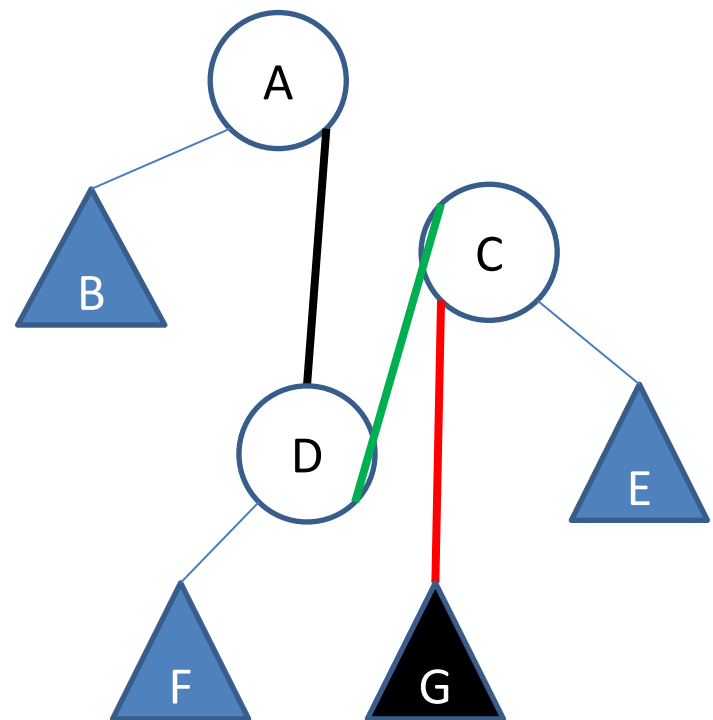


Durante a rotação

# Rotação direita-esquerda (3/5)



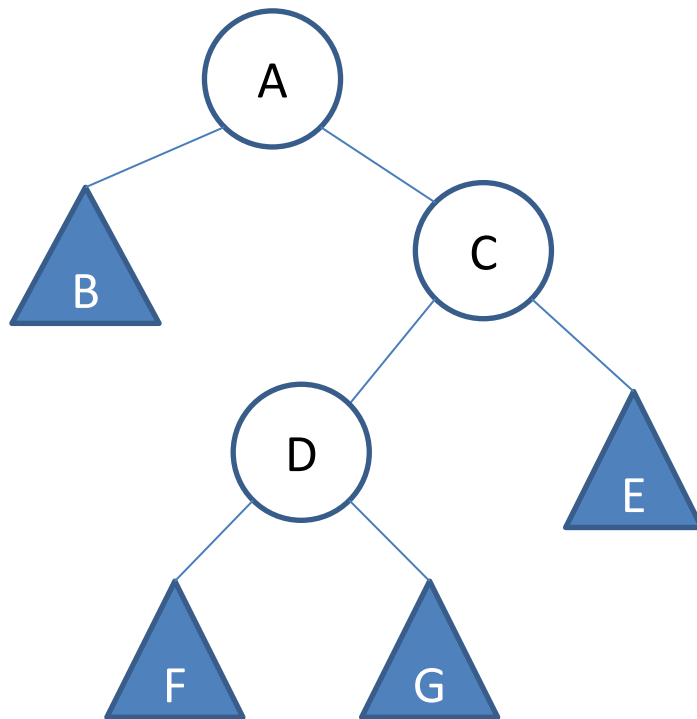
Antes da rotação



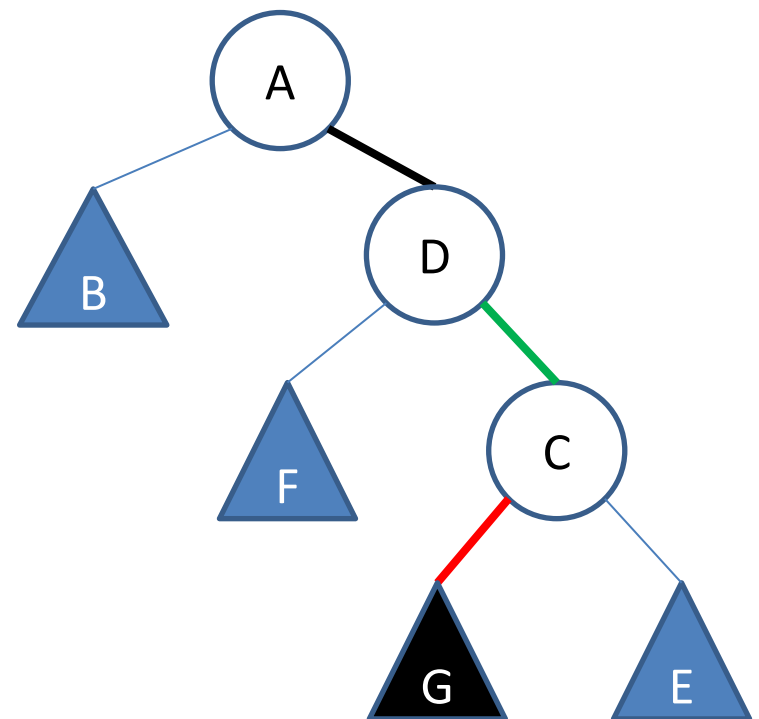
Durante a rotação



# Rotação direita-esquerda (3/5)

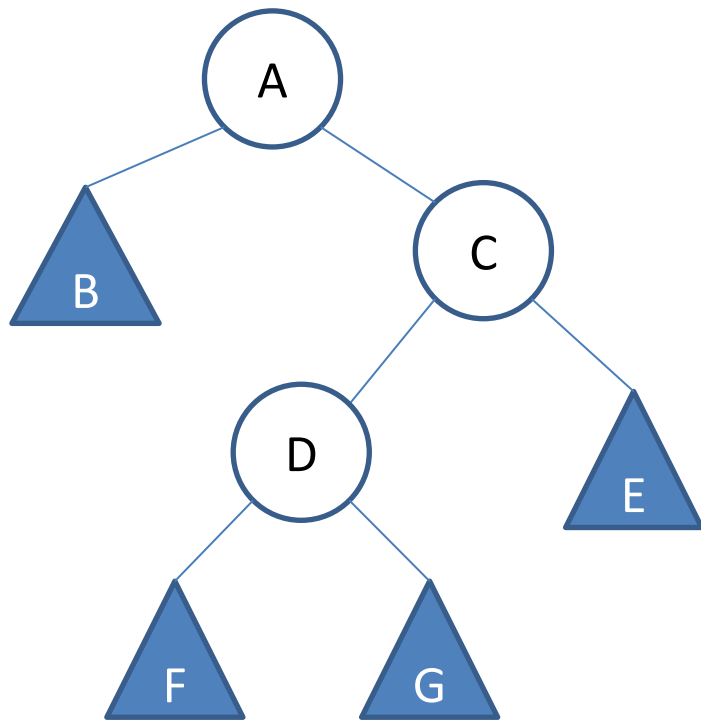


Antes da rotação

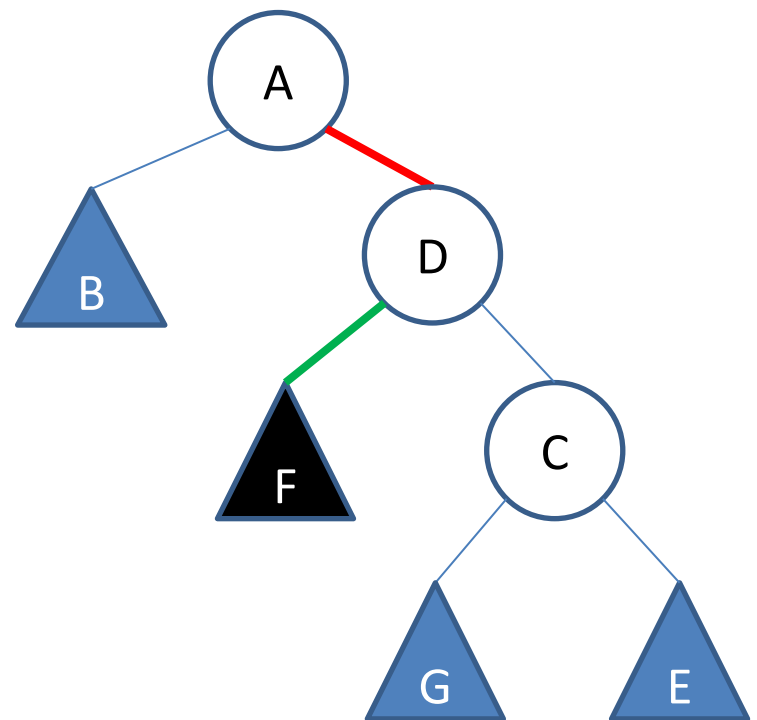


Durante a rotação

# Rotação direita-esquerda (3/5)

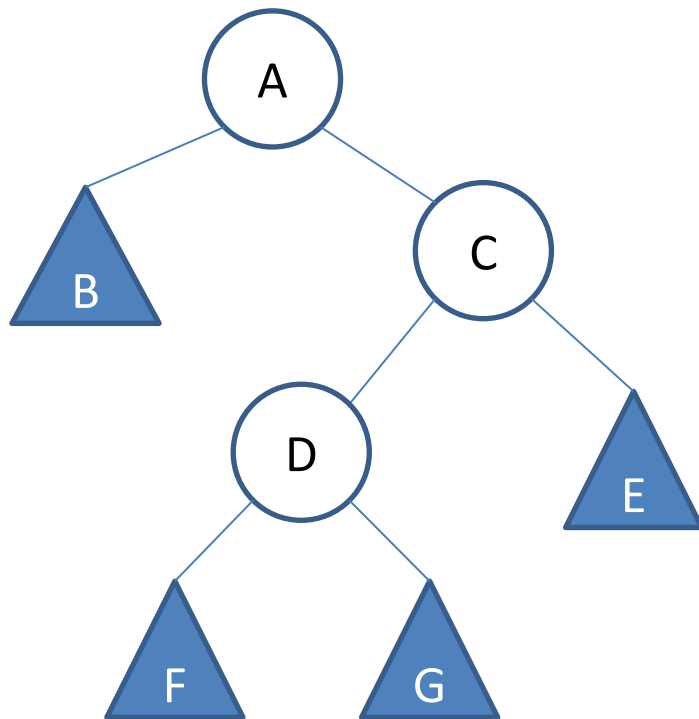


Antes da rotação

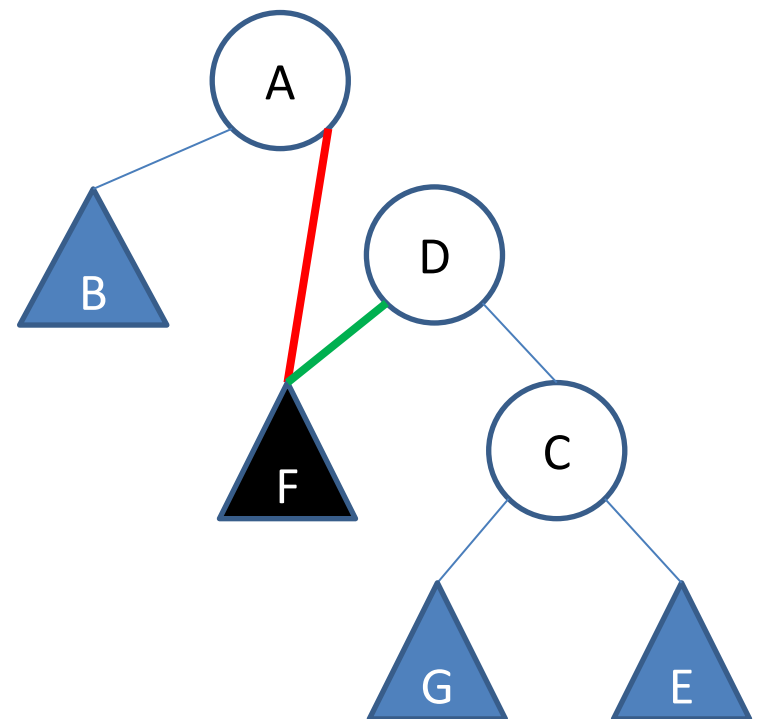


Durante a rotação

# Rotação direita-esquerda (4/5)

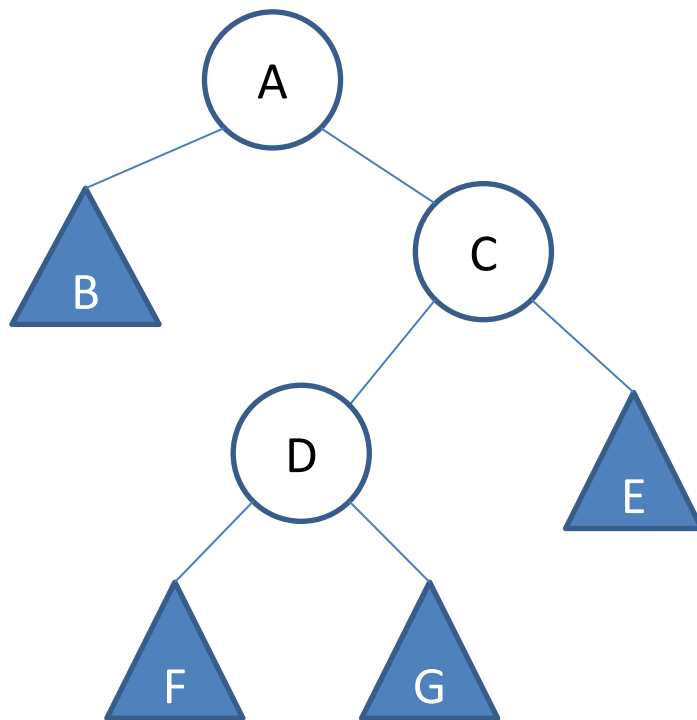


Antes da rotação

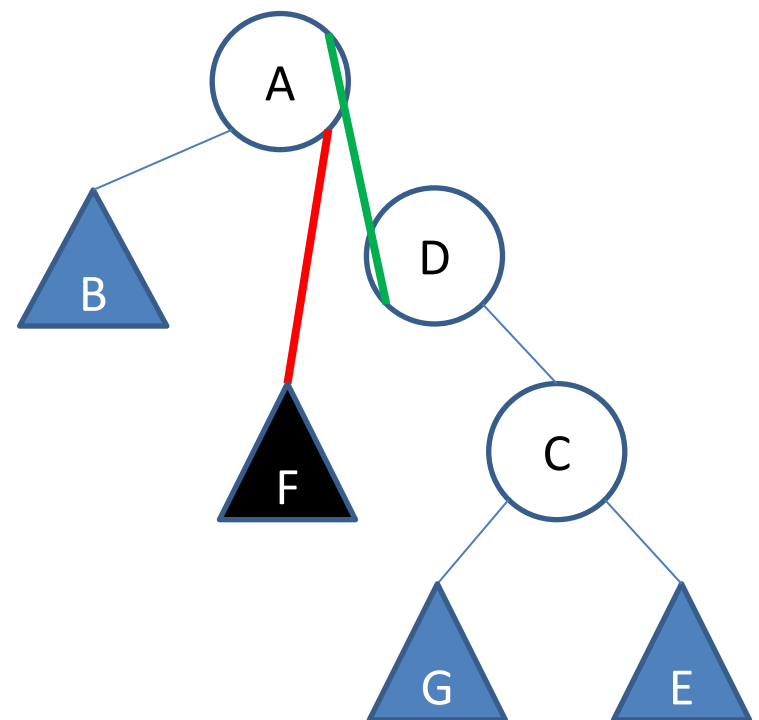


Durante a rotação

# Rotação direita-esquerda (5/5)

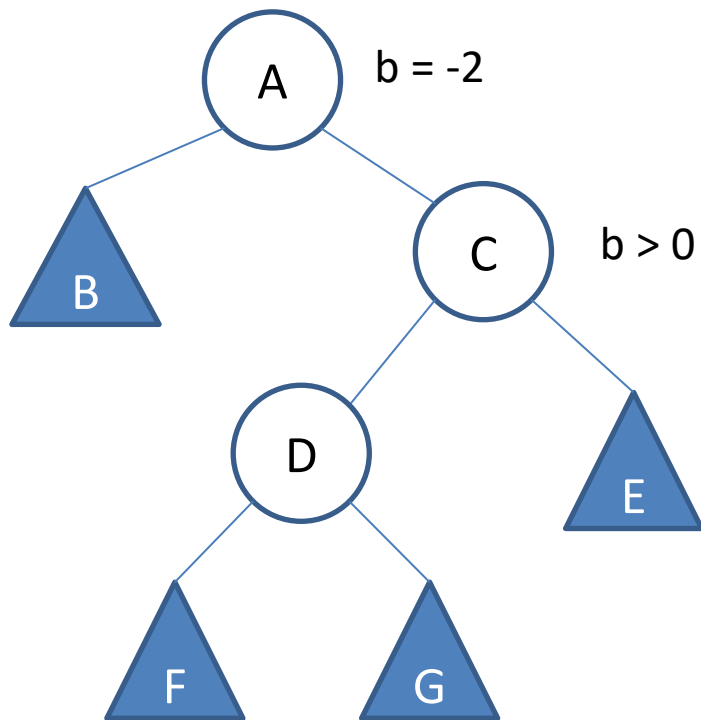


Antes da rotação

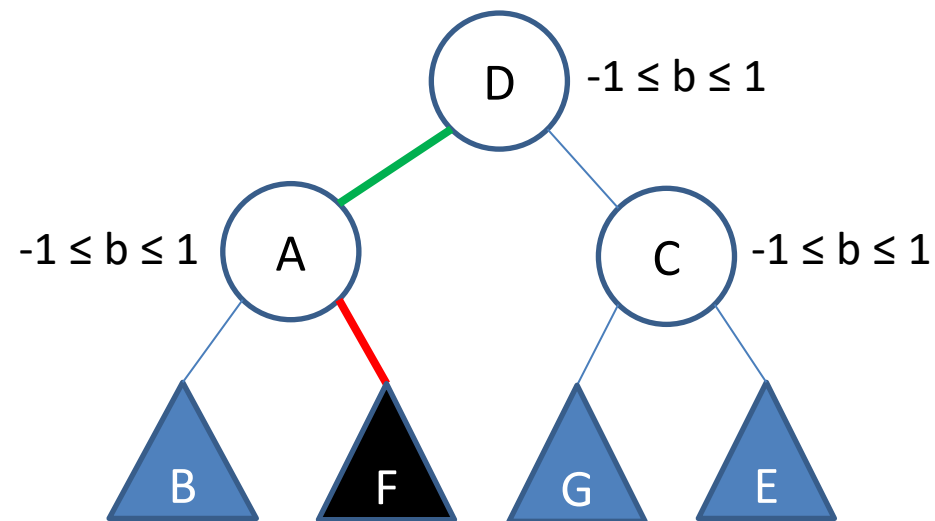


Durante a rotação

# Rotação direita-esquerda (5/5)



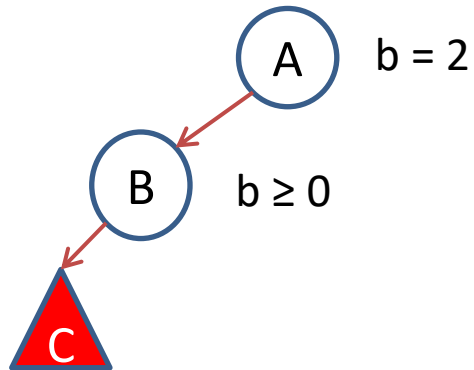
Antes da rotação



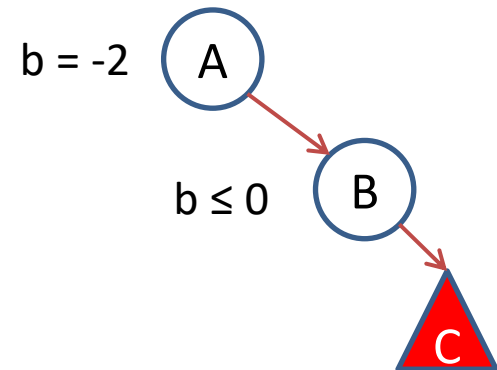
Depois da rotação

# **INSERÇÕES NA ÁRVORE AVL**

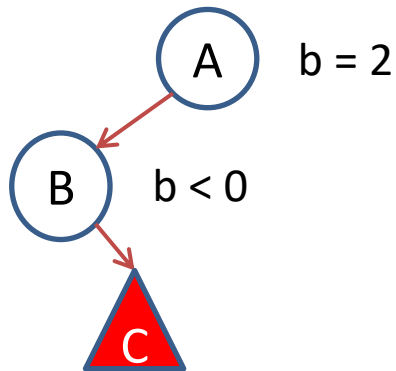
# Local da inserção sugere rotação



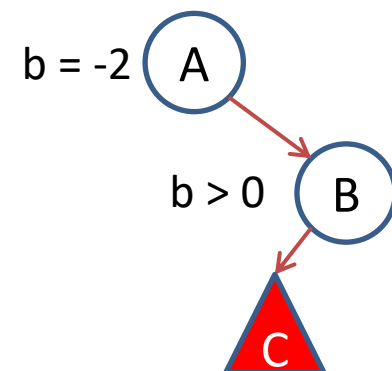
Rotação simples à direita



Rotação simples à esquerda

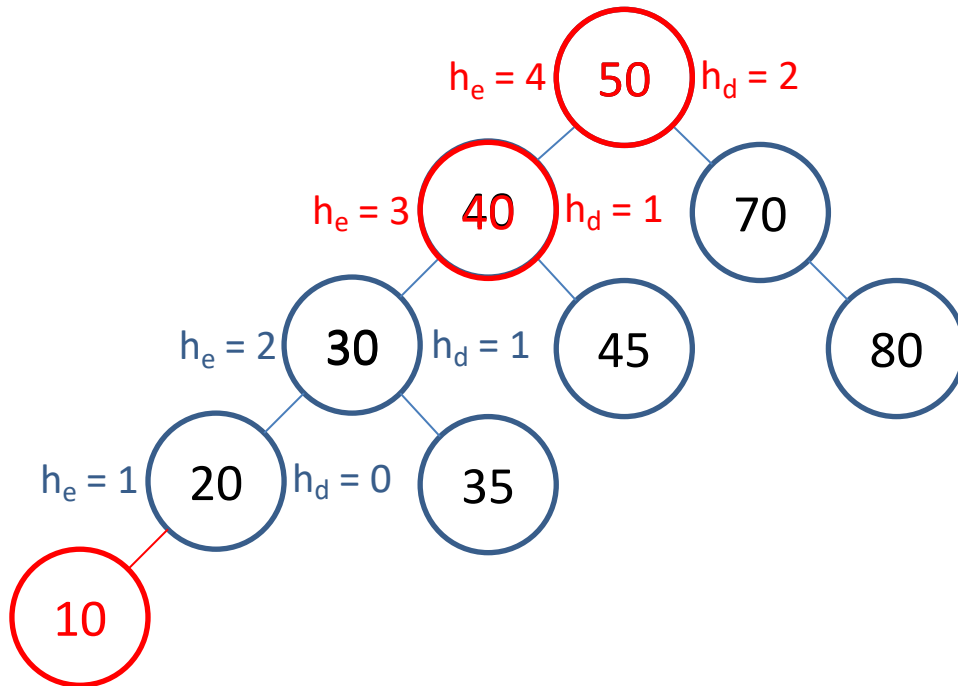


Rotação esquerda-direita



Rotação direita-esquerda

# Exemplo #1: inserção subárv. externa esq.

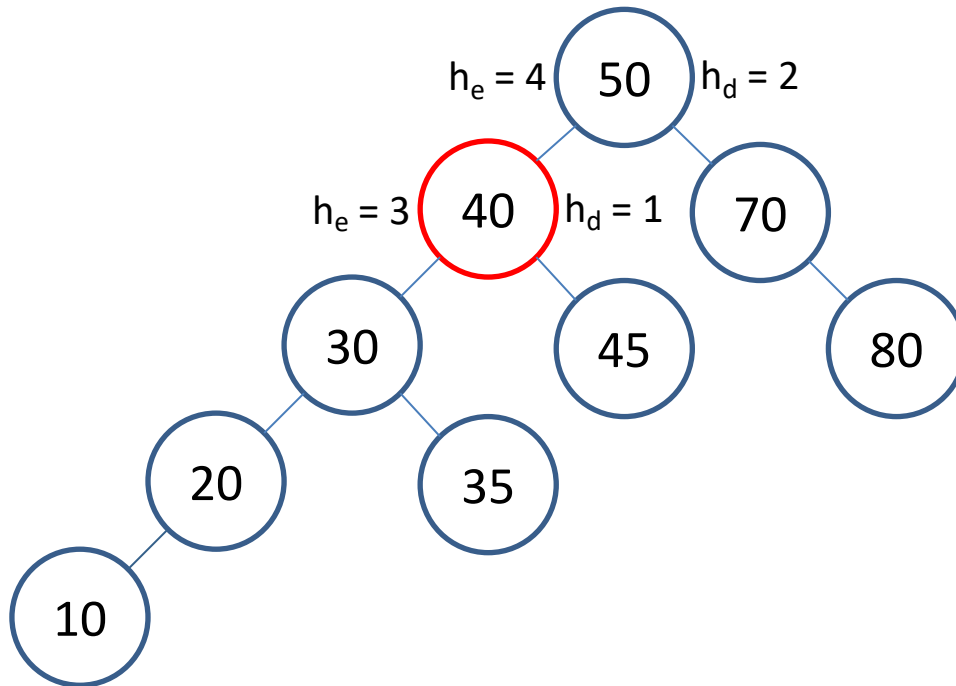


1º) Vamos inserir o nó 10.

**Resultado:** desbalanceou os nós 40 e 50.



# Exemplo #1: inserção subárv. externa esq.

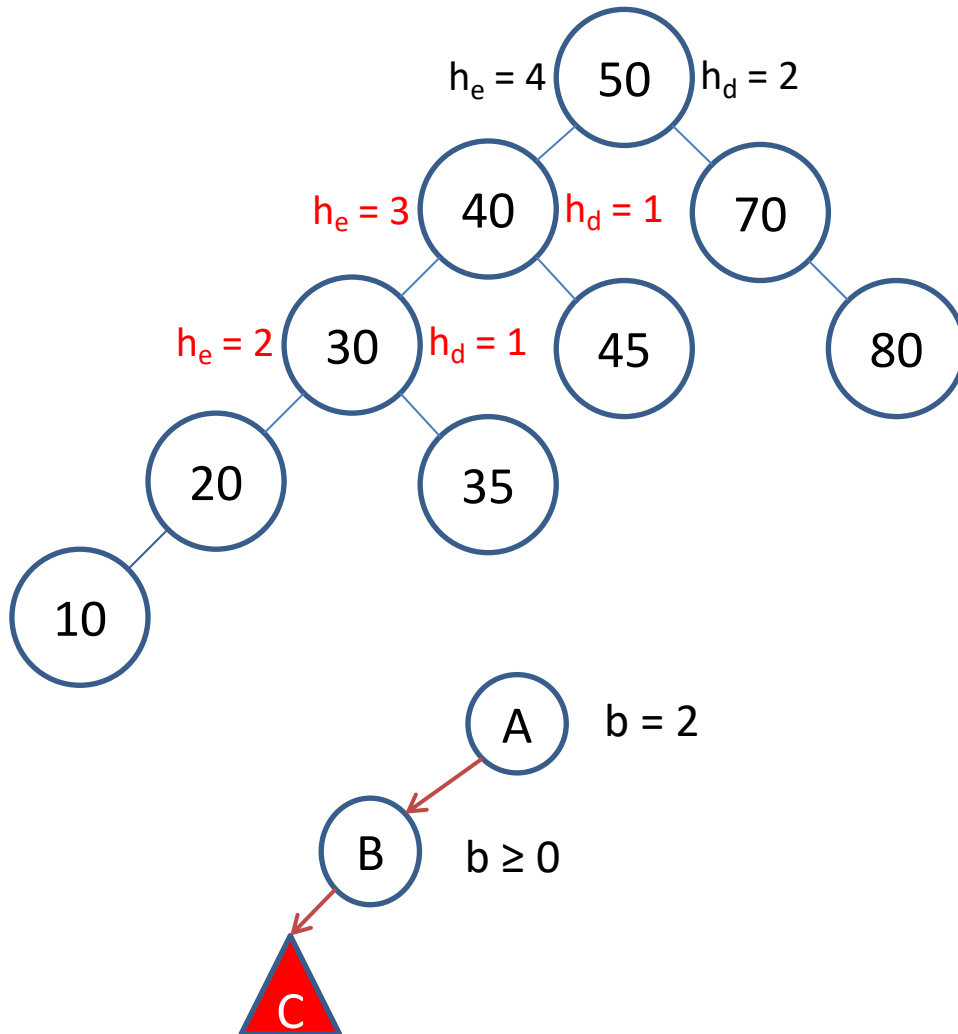


2º) Vamos balancear o nó 40.

**Por que o nó 40 e não o 50?**

Porque a idéia é que ao ajustarmos as subárvores esquerda e direita do nó 40, teremos automaticamente ajustado a subárvore esquerda do nó 50 também.

# Exemplo #1: inserção subárv. externa esq.

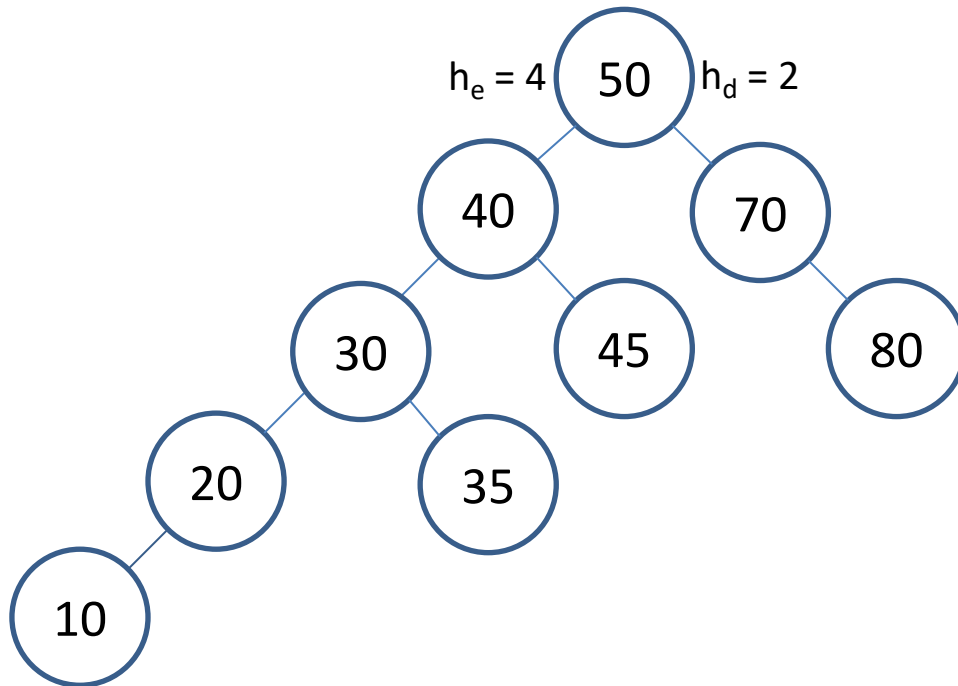


3º) Aplicar rotação simples à direita para balancear o nó 40.

**Por que rotação à direita?**  
Porque  $h_e > h_d$  para o nó 40. Logo a subárvore esquerda do nó 40 está mais pesada que a subárvore direita do nó 40.

**Por que rotação simples?**  
Porque  $h_e > h_d$  para o nó 30 (nó filho esquerdo do nó 40, que é a subárvore mais pesada).  
Desse fato se conclui que o novo nó (nó 10) foi inserido na subárvore esquerda do nó 30.

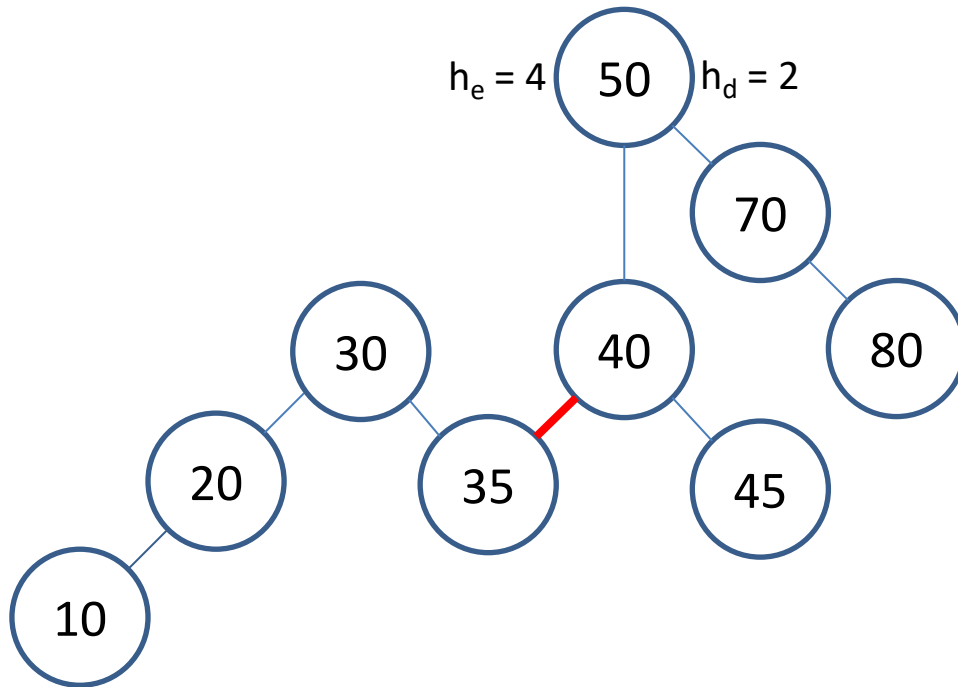
# Exemplo #1: inserção subárv. externa esq.



3º) Rotação simples à direita do nó 40 (primeiro nó que ficou desbalanceado).



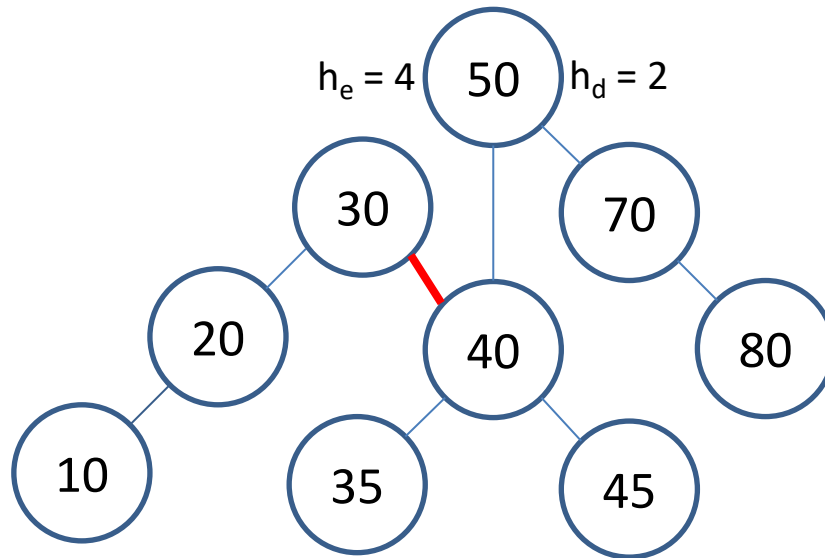
# Exemplo #1: inserção subárv. externa esq.



3ª) Rotação simples à direita do nó 40 (primeiro nó que ficou desbalanceado).

➤ Filho a direita do nó 30 passa a ser filho a esquerda do nó 40.

# Exemplo #1: inserção subárv. externa esq.

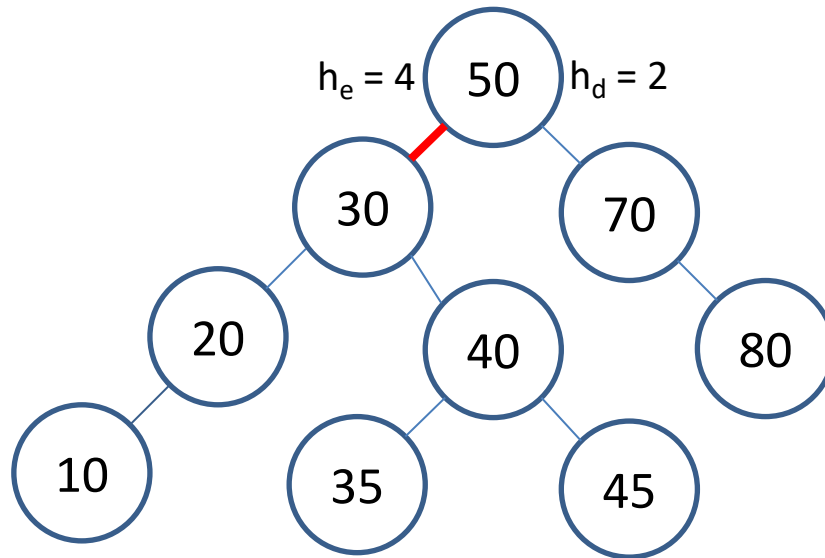


3º) Rotação simples à direita do nó 40 (primeiro nó que ficou desbalanceado).

➤ Filho a esquerda do nó 40 passa a ser o filho a direita do nó 30.

➤ Filho a direita do nó 30 passa a ser o nó 40.

# Exemplo #1: inserção subárv. externa esq.



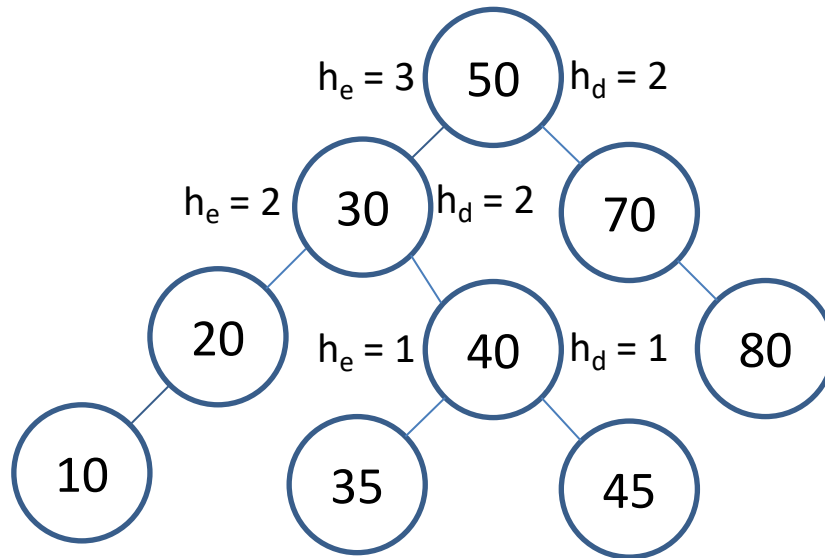
3º) Rotação simples à direita do nó 40 (primeiro nó que ficou desbalanceado).

➤ Filho a esquerda do nó 40 passa a ser o filho a direita do nó 30.

➤ Filho a direita do nó 30 passa a ser o nó 40.

➤ Filho a esquerda do nó 50 passa a ser o nó 30.

# Exemplo #1: inserção subárv. externa esq.

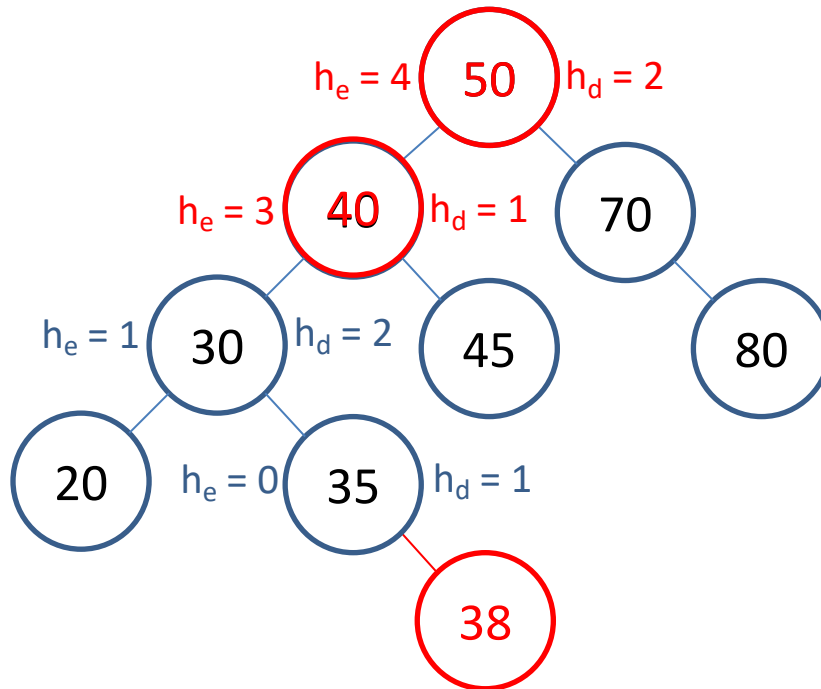


**E resolveu os desbalanceamentos?**

Sim, todas as diferenças de alturas agora são menores que 2.

- Continua uma árvore de pesquisa binária?  
Sim. Percurso “em ordem”:  
**10 20 30 35 40 45 50 70 80**

# Exemplo #3: inserção subárv. interna esq.

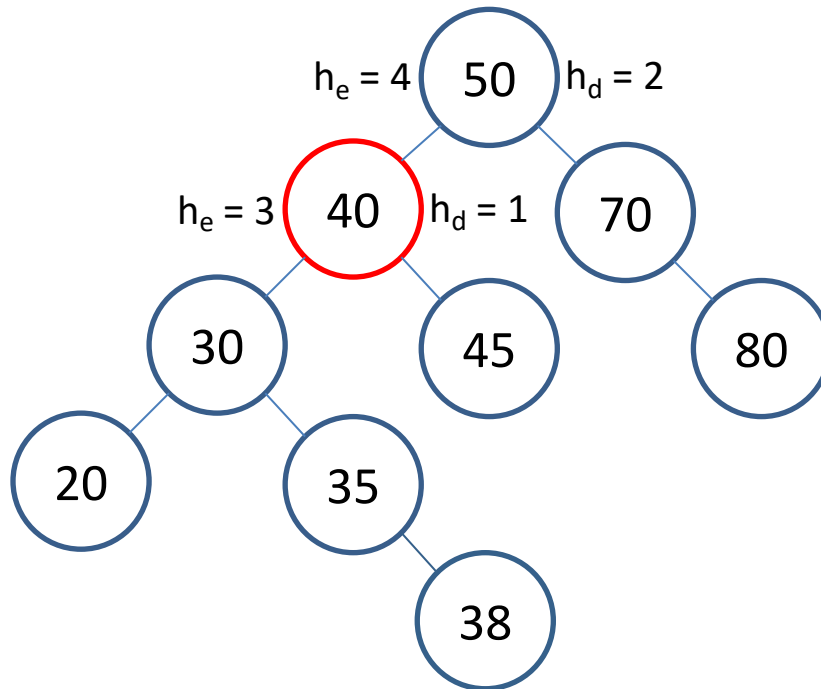


1º) Vamos inserir o nó 38.

**Resultado:** desbalanceou os nós 40 e 50.



## Exemplo #3: inserção subárv. interna esq.

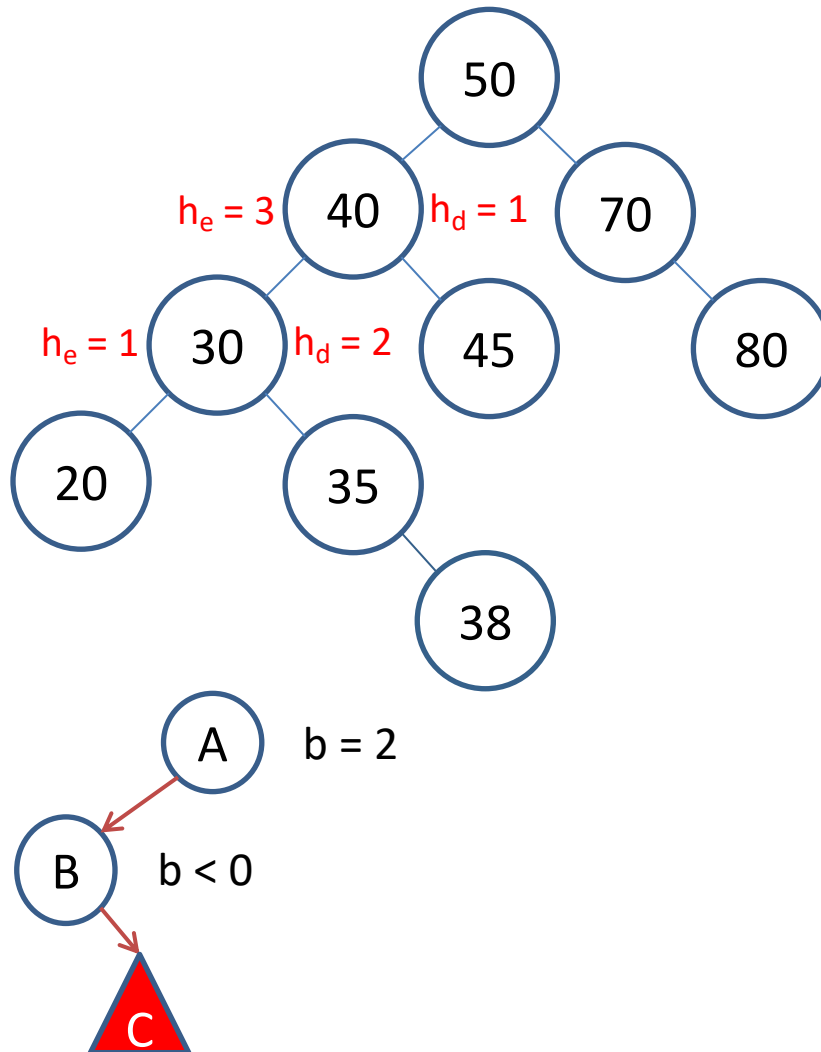


2º) Vamos balancear o nó 40.

**Por que o nó 40 e não o 50?**

Porque a idéia é que ao ajustarmos as subárvores esquerda e direita do nó 40, teremos automaticamente ajustado a subárvore esquerda do nó 50 também.

# Exemplo #3: inserção subárv. interna esq.

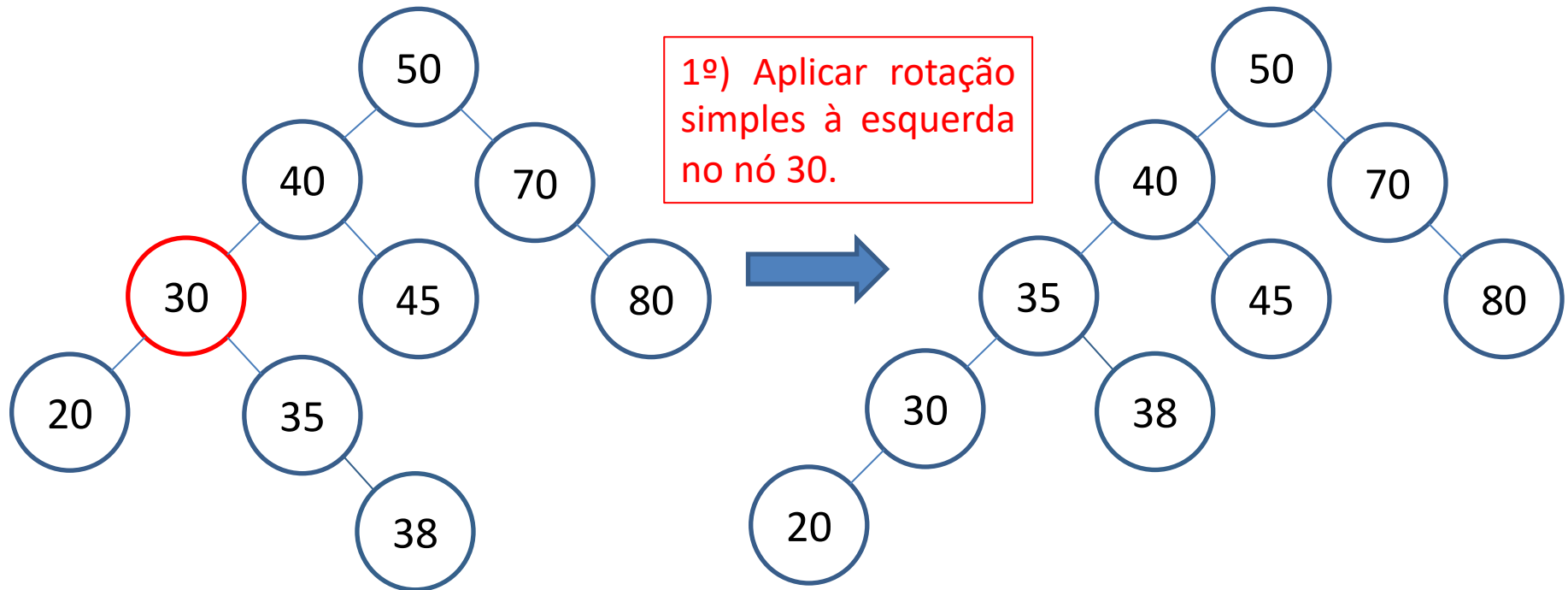


3º) Aplicar rotação dupla à direita para balancear o nó 40.

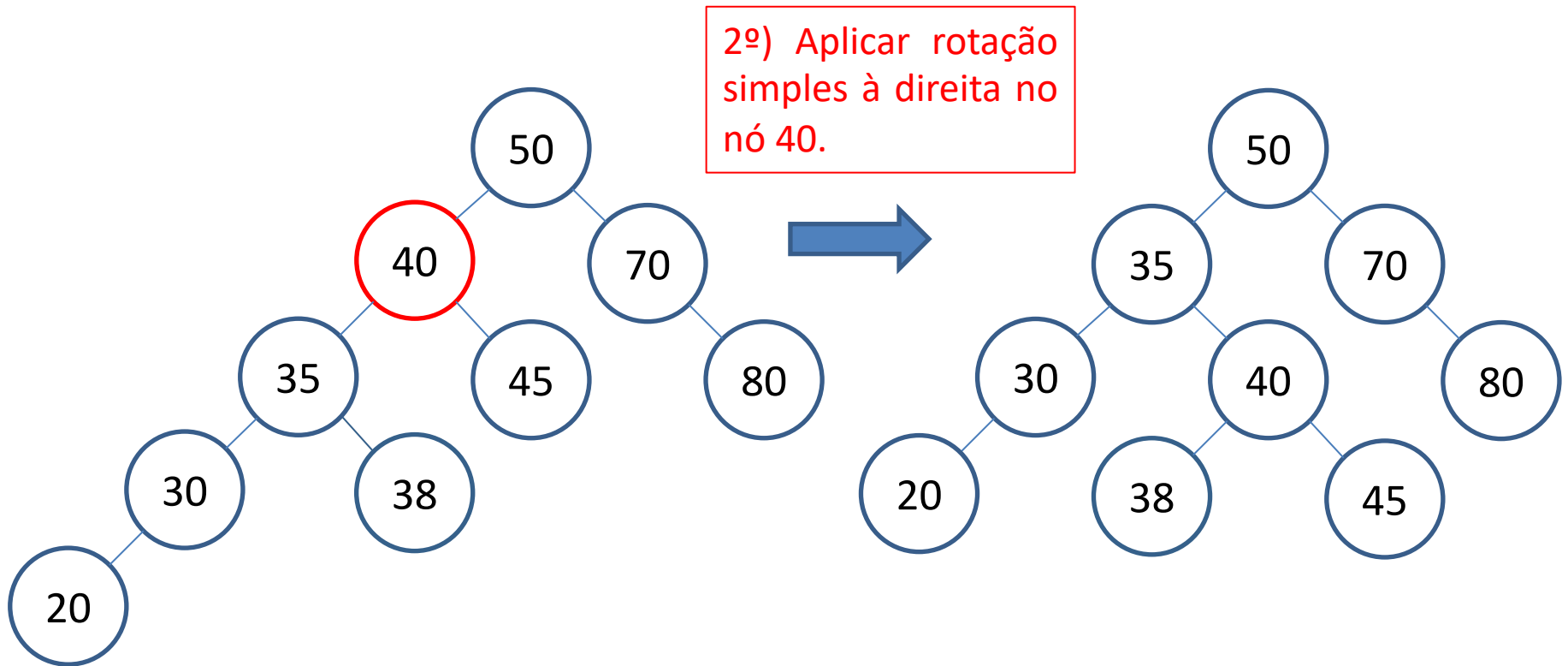
**Por que rotação à direita?**  
Porque  $h_e > h_d$  para o nó 40. Logo a subárvore esquerda do nó 40 está mais pesada que a subárvore direita do nó 40.

**Por que rotação dupla?**  
Porque  $h_e < h_d$  para o nó 30 (nó filho esquerdo do nó 40, que é a subárvore mais pesada).  
Desse fato se conclui que o novo nó (nó 38) foi inserido na subárvore direita do nó 30.

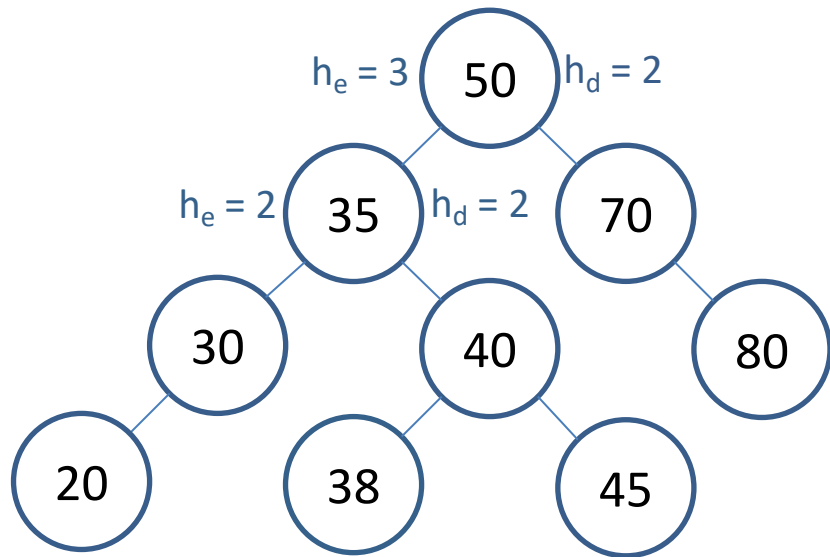
## Exemplo #3: inserção subárv. interna esq.



## Exemplo #3: inserção subárv. interna esq.



## Exemplo #3: inserção subárv. interna esq.



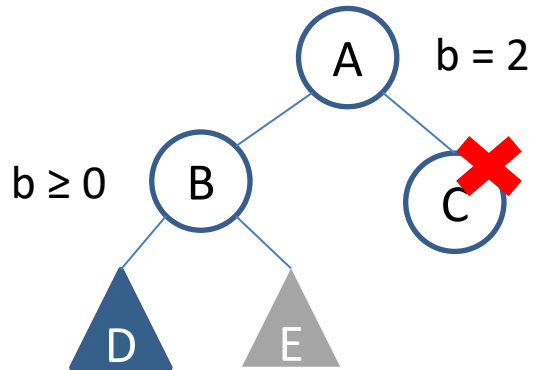
**E resolveu os desbalanceamentos?**

Sim, todas as diferenças de alturas agora são menores que 2.

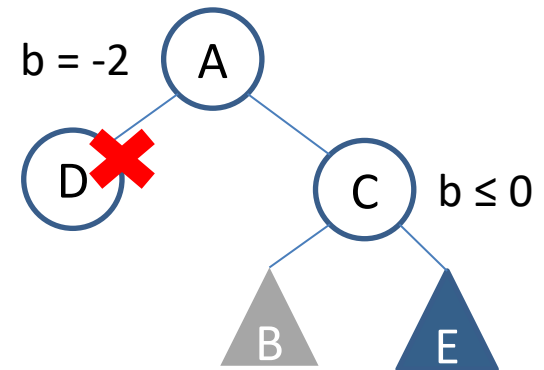
- Continua uma árvore de pesquisa binária?  
Sim. Percurso “em ordem”:  
**20 30 35 38 40 45 50 70 80**

# **DELEÇÕES NA ÁRVORE AVL**

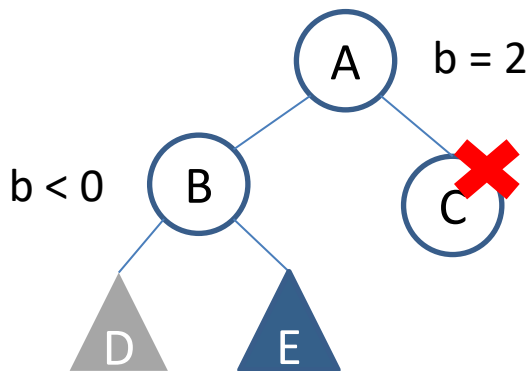
# Resultado da **deleção** sugere rotação



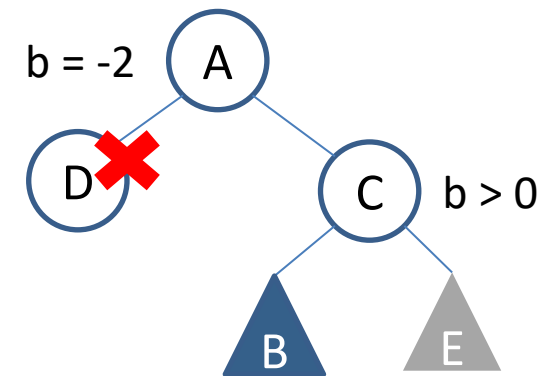
Rotação simples à direita



Rotação simples à esquerda

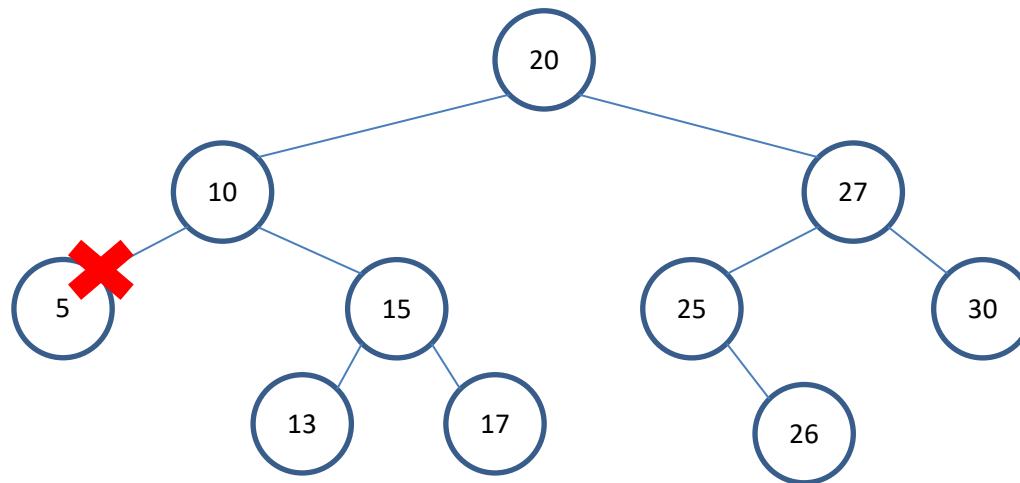


Rotação esquerda-direita



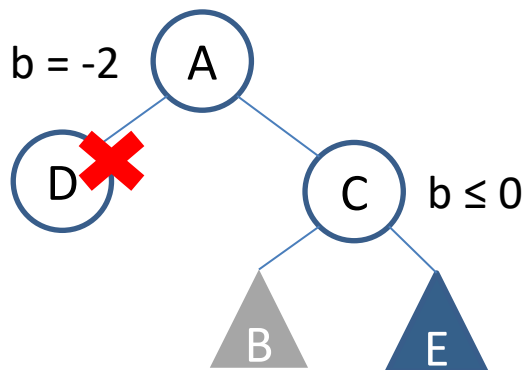
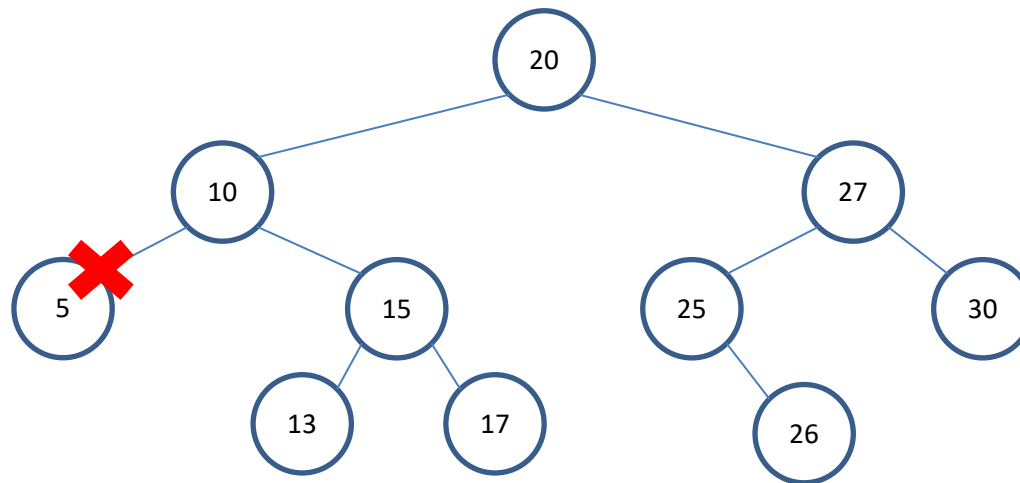
Rotação direita-esquerda

# Exemplo de deleção



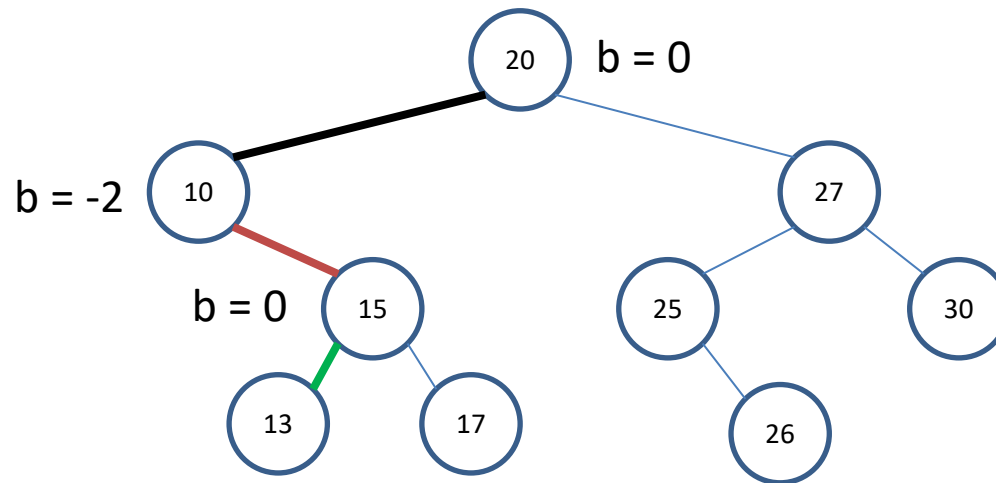


# Exemplo de deleção

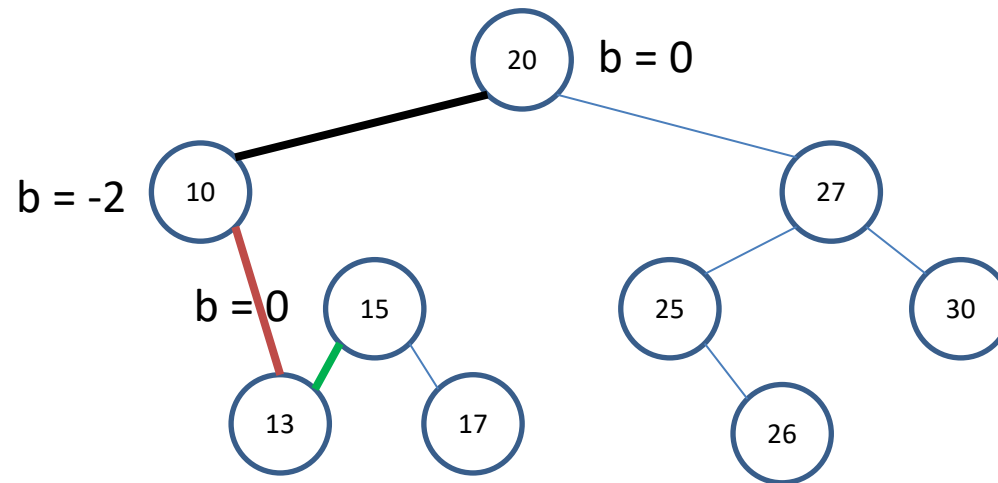


Rotação simples à esquerda

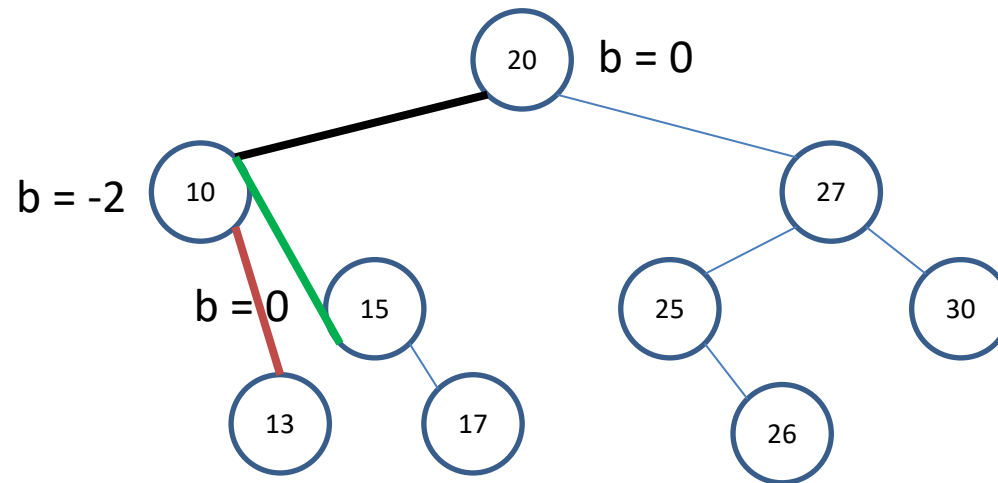
# Exemplo de deleção



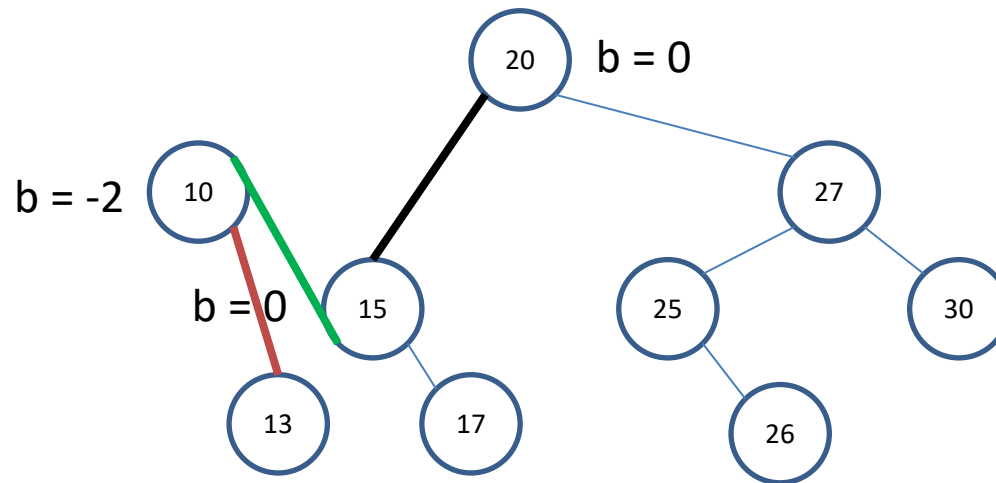
# Exemplo de deleção



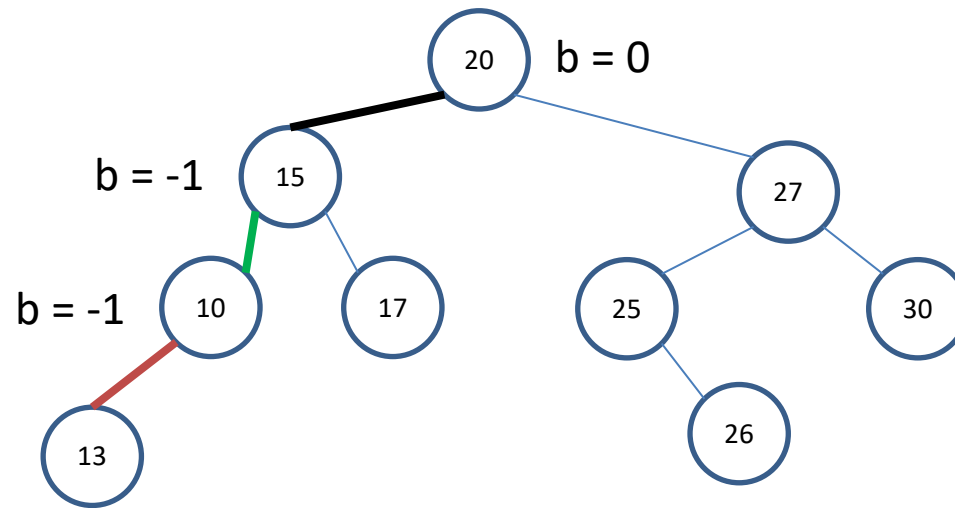
# Exemplo de deleção



# Exemplo de deleção



# Exemplo de deleção



# Resumo

- AVL são árvores binárias parc. Balanceadas
- Fator  $b \geq 1$  ou  $b \leq -1$  exige rotações
- Quatro tipos de rotação
  - Simples à direita (aka Right-Right)
  - Simples à Esquerda (aka Left-Left)
  - Esquerda, depois direita (aka Left-Right)
  - Direita, depois esquerda (aka Right-Left)