

Árvores AVL e SPLAY

Profa. Franciny Medeiros

Disciplina de Estrutura de Dados 2

Bacharelado em Ciências da Computação – UFJ

Roteiro

- Árvores balanceadas

- AVL: fator de balanceamento, rotações simples e dupla, inserção e remoção.

- SPLAY: objetivo, rotações, pesquisa, inserção e remoção.

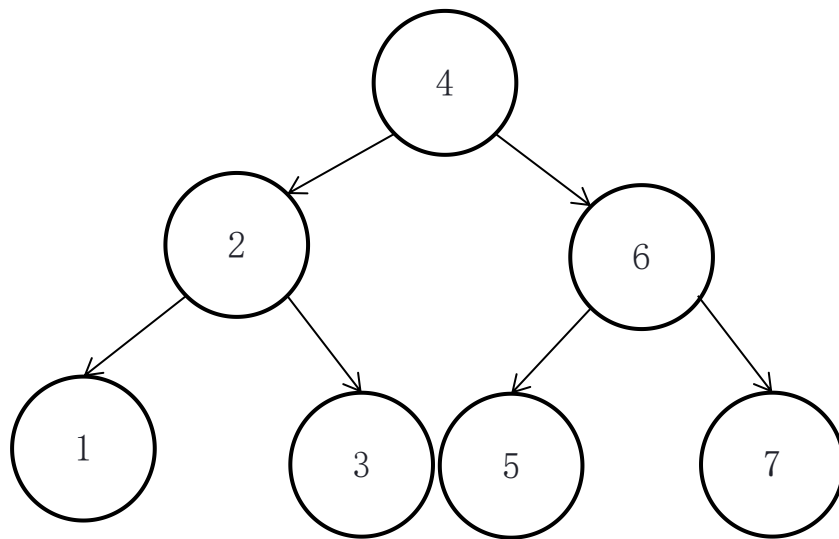
Árvores Balanceadas

- ❑ As árvores são projetadas para um acesso rápido à informação.
- ❑ A eficiência da busca em uma árvore binária depende da distribuição de seus elementos na estrutura.
- ❑ Tempo de pesquisa $O(\log n)$.

Árvores Balanceadas

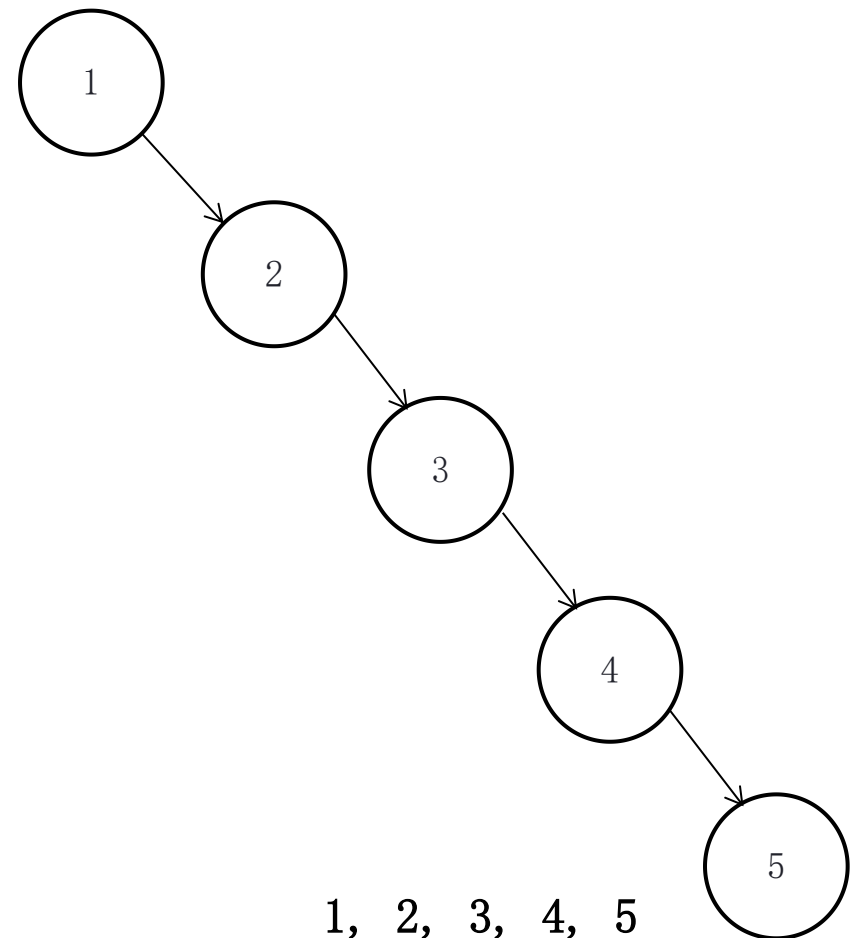
- ❑ Porém, operações de inserção e remoção podem degenerar a árvore.
- ❑ Tempo de pesquisa em uma árvore não balanceada pode ser dado por $O(n)$.

Árvores Balanceadas



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

$O(\log n)$



1, 2, 3, 4, 5

$O(n)$

Árvores Balanceadas

- ❑ Solução para o problema de balanceamento:
 - ❑ rebalancear a árvore no momento de inserção ou remoção de algum elemento.
- ❑ Árvores balanceadas:
 - ❑ árvores AVL.
 - ❑ Árvores Splay.
 - ❑ Árvores rubro-negra.
 - ❑ Árvore 2-3-4.

Árvores Balanceadas

- ❑ Solução para o problema de balanceamento:
 - ❑ rebalancear a árvore no momento de inserção ou remoção de algum elemento.
- ❑ Árvores balanceadas:
 - ❑ árvores AVL.
 - ❑ Árvores Splay.
 - ❑ Árvores rubro-negra.
 - ❑ Árvore 2-3-4.

Árvores Balanceadas: AVL

- ❑ Criada por Adelson – Velskii e Landis em 1962.
- ❑ Uma árvore binária onde as alturas das sub-árvores direita e esquerda de cada nó diferem de no máximo uma unidade.

Árvores Balanceadas: AVL

- Balanceamento é requerido nas operações de inserção e remoção.
- Para definir o balanceamento é utilizado um fator específico para os nós.

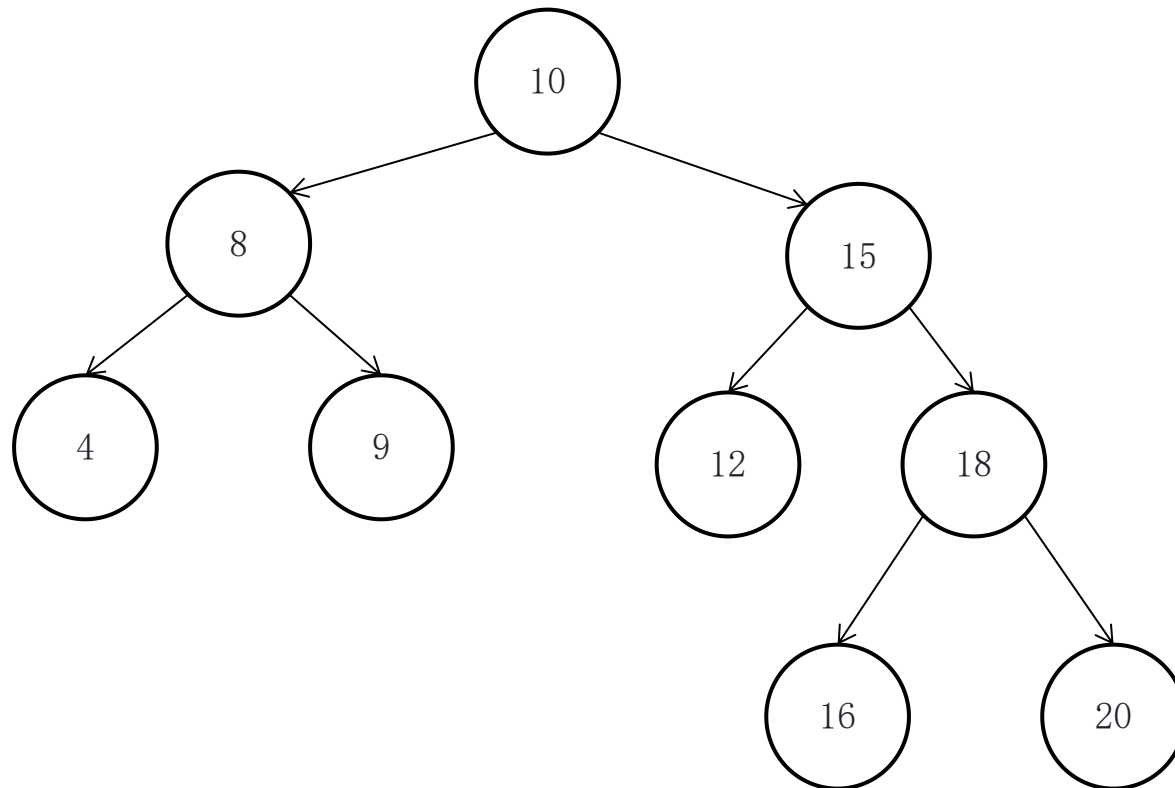
Árvores Balanceadas: AVL

□ Fator de balanceamento

altura sub-árvore **direita** – altura sub-árvore
esquerda

Árvores Balanceadas : AVL

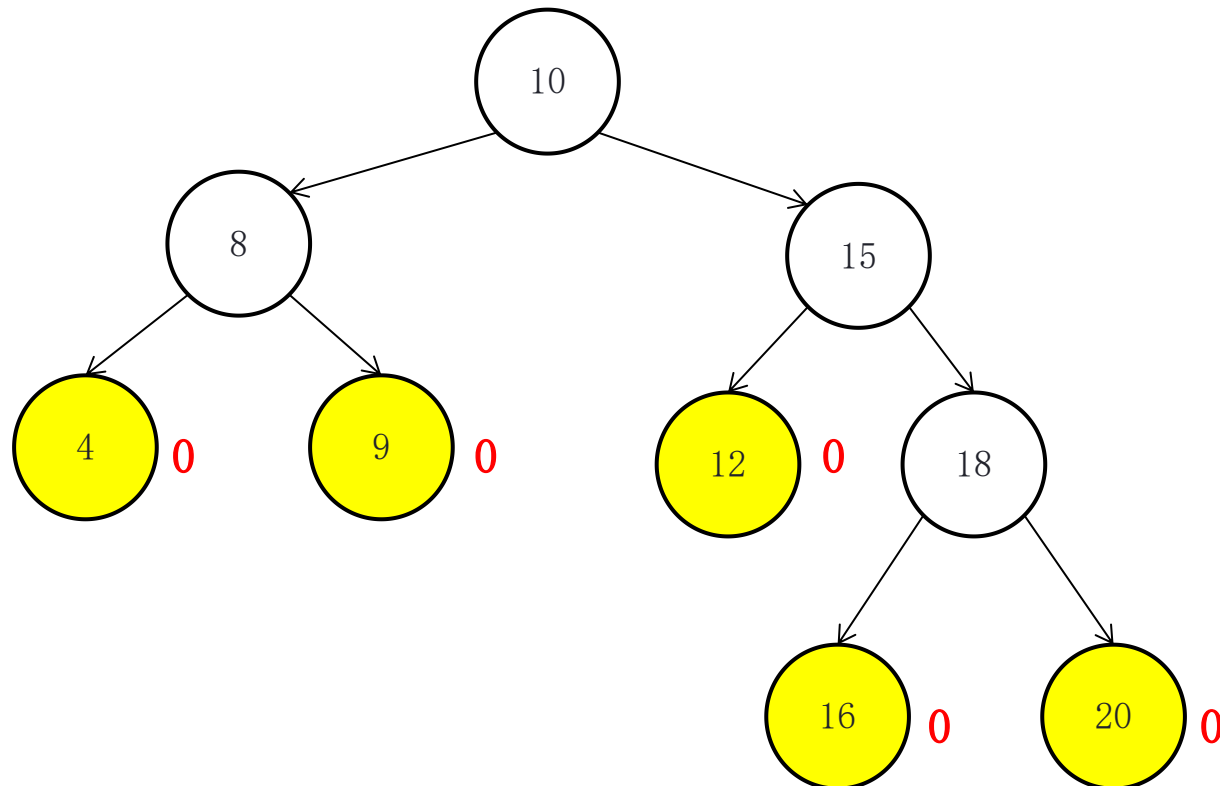
❏ Como calcular o fator de balanceamento



Árvores Balanceadas : AVL

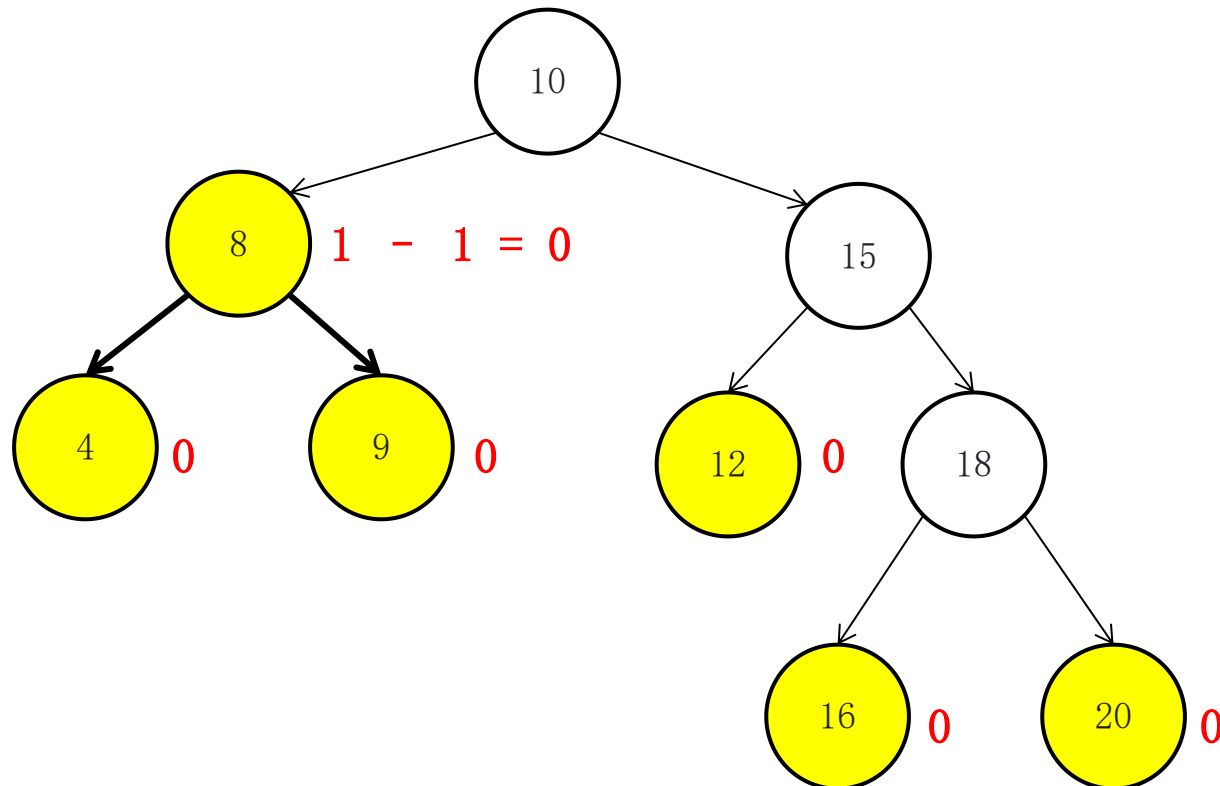
□ Como calcular o fator de balanceamento

hd - he



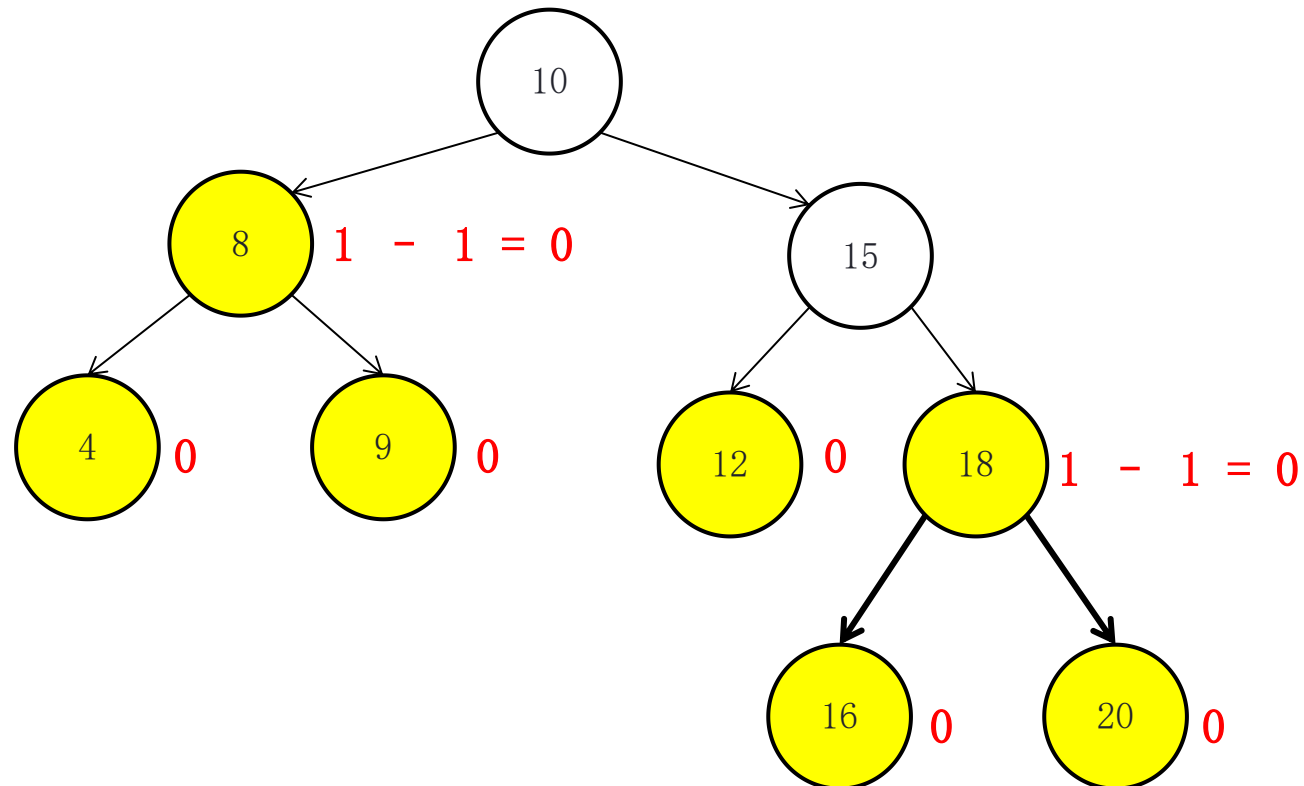
Árvores Balanceadas : AVL

❏ Como calcular o fator de balanceamento



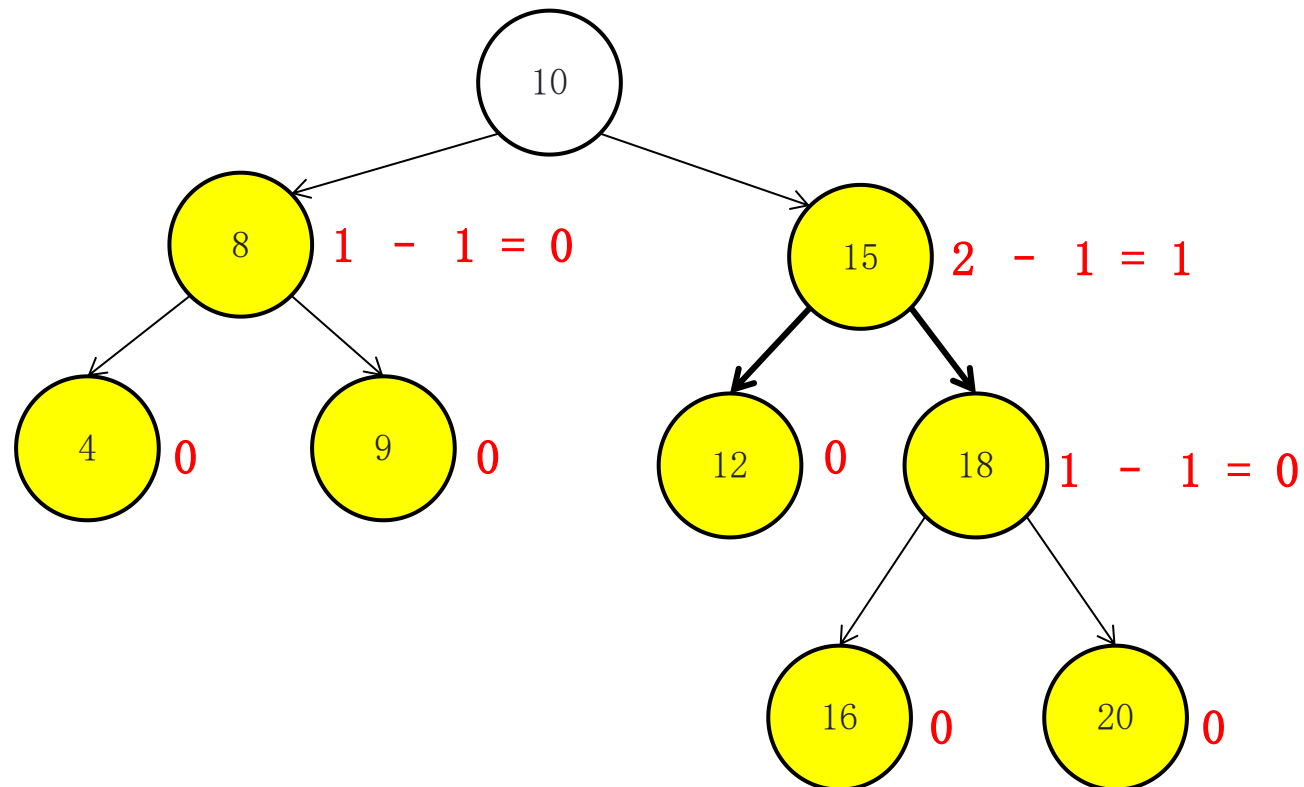
Árvores Balanceadas : AVL

❏ Como calcular o fator de balanceamento



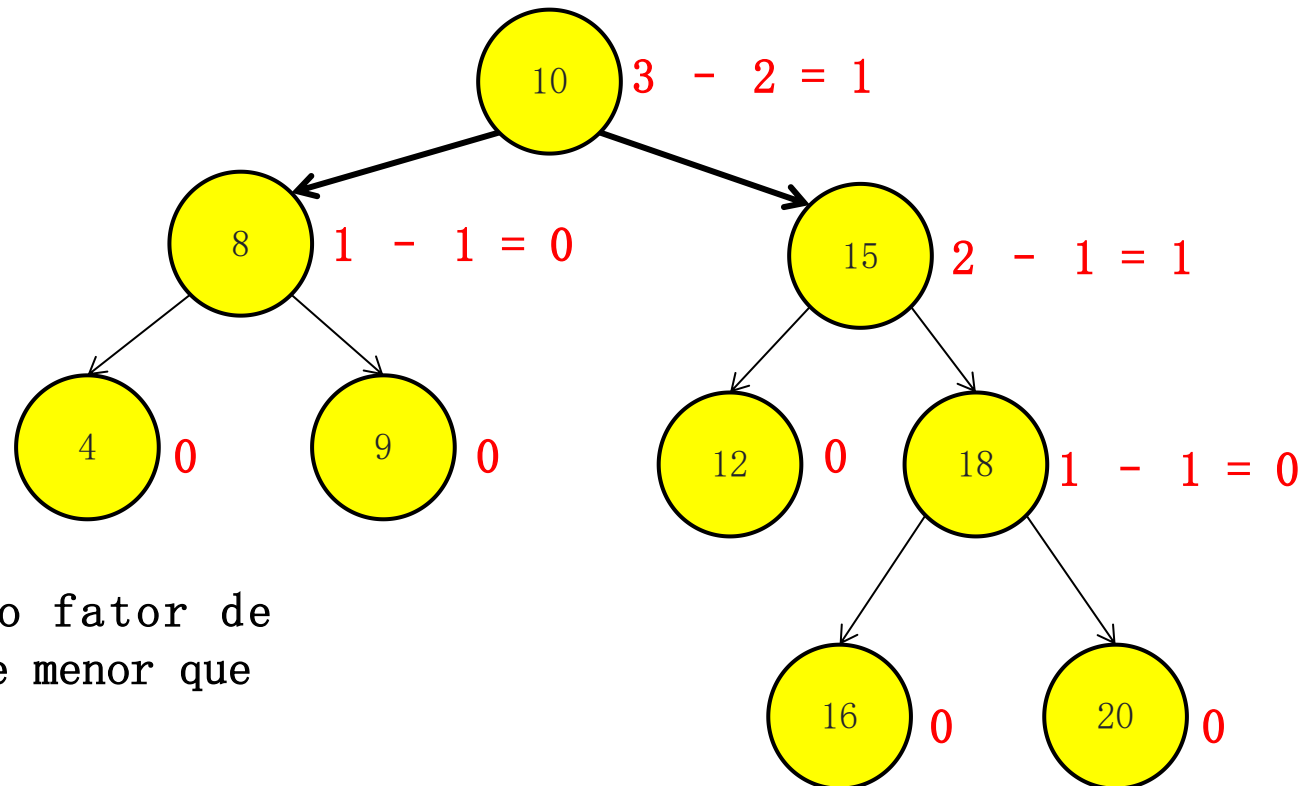
Árvores Balanceadas : AVL

❏ Como calcular o fator de balanceamento



Árvores Balanceadas : AVL

❏ Como calcular o fator de balanceamento



A ideia é manter o fator de balanceamento sempre menor que 2 e maior que -1

Árvores Balanceadas: AVL

- ❑ Para manter a árvore balanceada é necessário fazer uma transformação:
 - ❑ a árvore transformada continua sendo uma árvore binária;
 - ❑ a árvore transformada fique balanceada.

Árvores Balanceadas: AVL

- ❑ A transformação a ser feita na árvore é chamada de **rotação**.
- ❑ Rotação simples e Rotação dupla.

Árvores Balanceadas: AVL

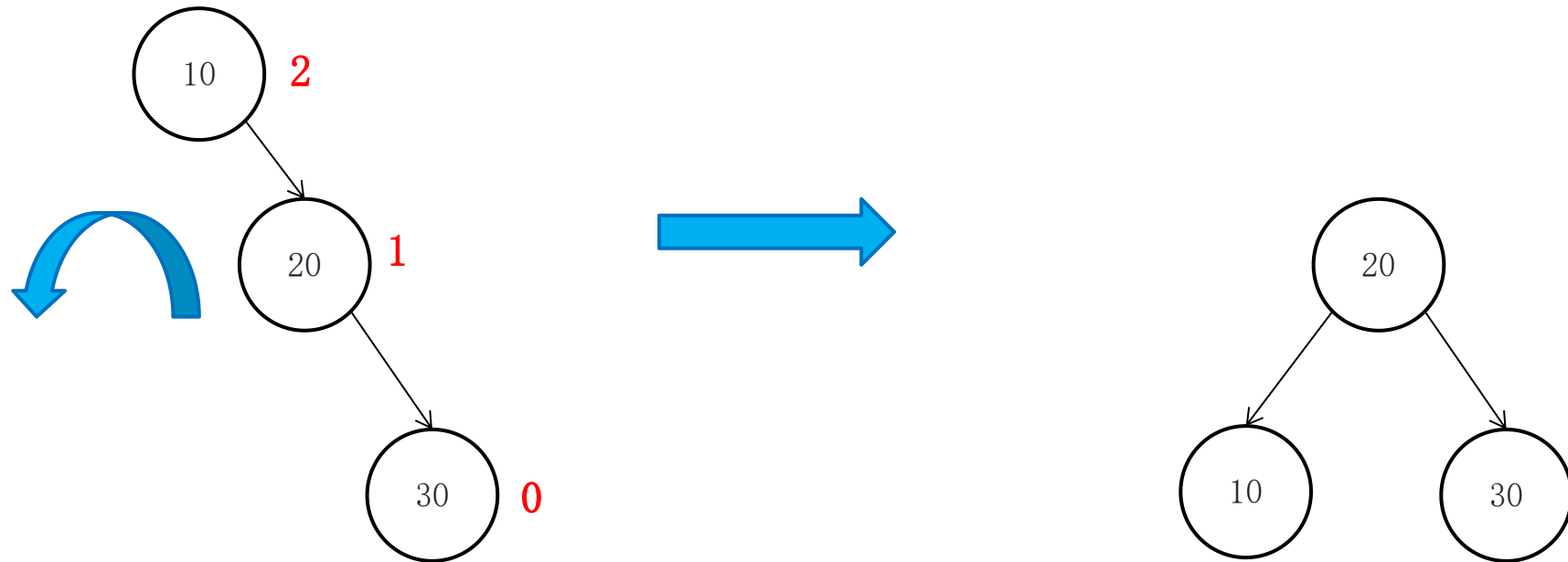
❑ Rotação simples

- ❑ ocorre quando um nó está desbalanceado e seu filho estiver no mesmo sentido da inclinação.

- ❑ Rotação à esquerda e Rotação à direita

Árvores Balanceadas: AVL

□ Rotação simples à esquerda



Rotação simples à esquerda

```
FUNÇÃO rotacaoEsquerda(z)
```

```
  y = z.direito
```

```
  T2 = y.esquerdo
```

```
  y.esquerdo = z
```

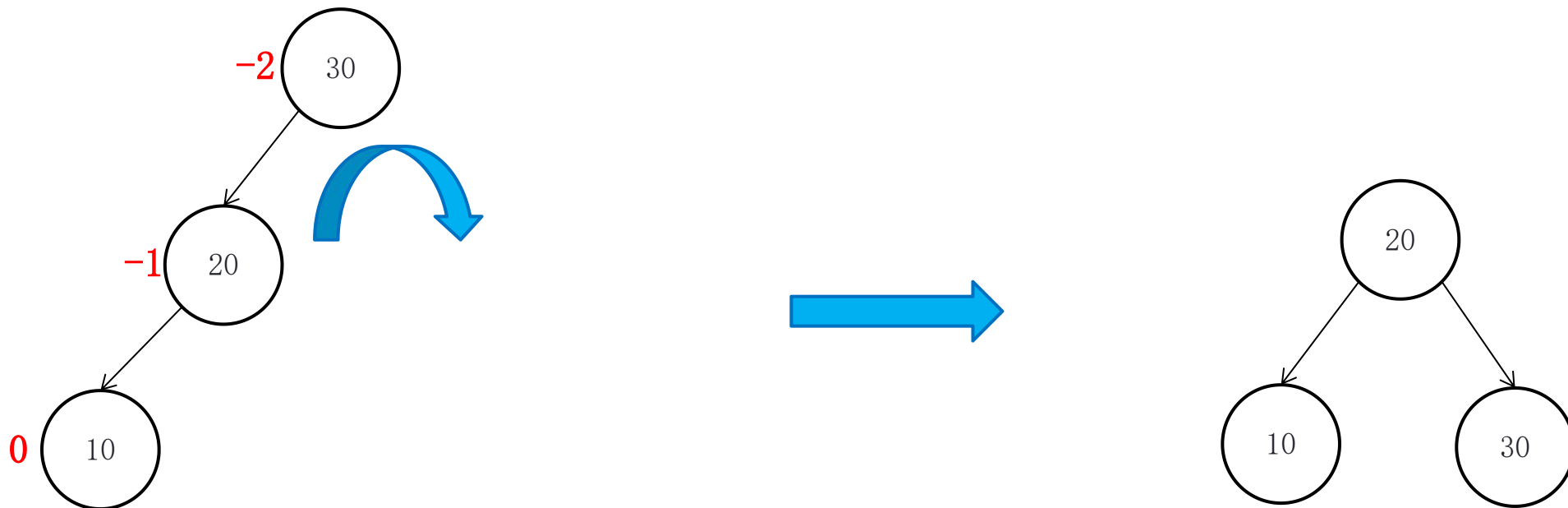
```
  z.direito = T2
```

```
  RETORNE y
```

```
FIM FUNÇÃO
```

Árvores Balanceadas: AVL

❑ Rotação simples à direita



Rotação simples à direita

```
FUNÇÃO rotacaoDireita(z)
```

```
  y = z.esquerdo
```

```
  T3 = y.direito
```

```
  y.direito = z
```

```
  z.esquerdo = T3
```

```
  RETORNE y
```

```
FIM FUNÇÃO
```

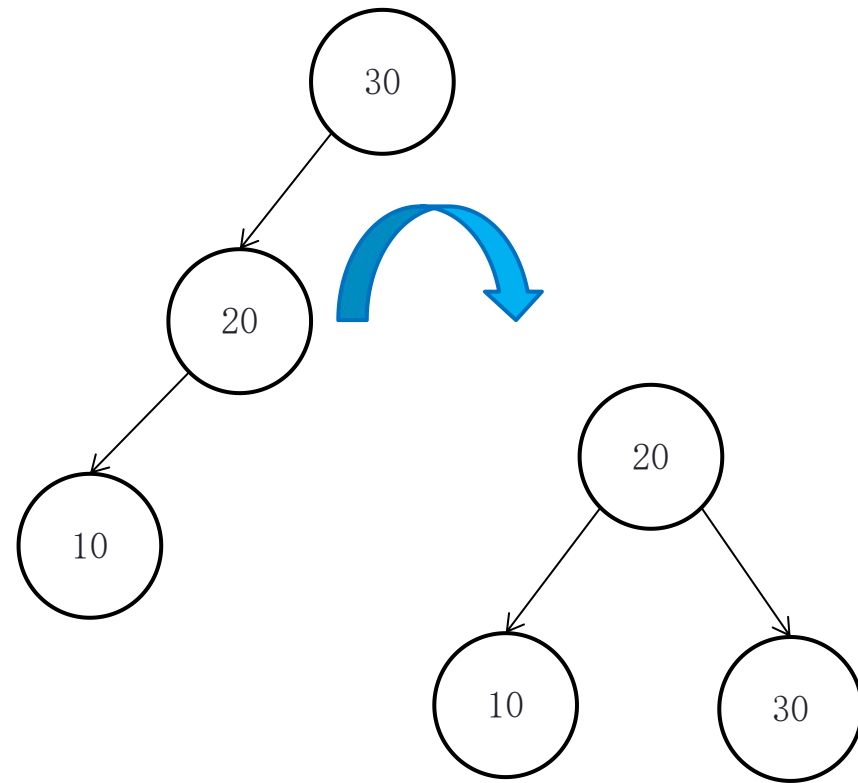
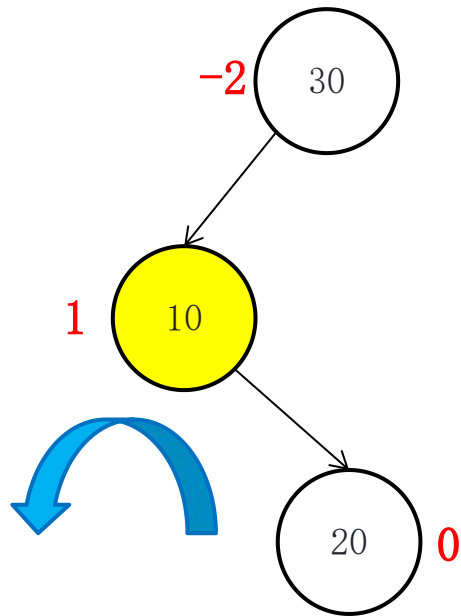
Árvores Balanceadas: AVL

❑ Rotação dupla

- ❑ Uma rotação dupla ocorre quando um nó estiver desbalanceado e seu filho estiver inclinado no sentido inverso ao pai, formando um "joelho".

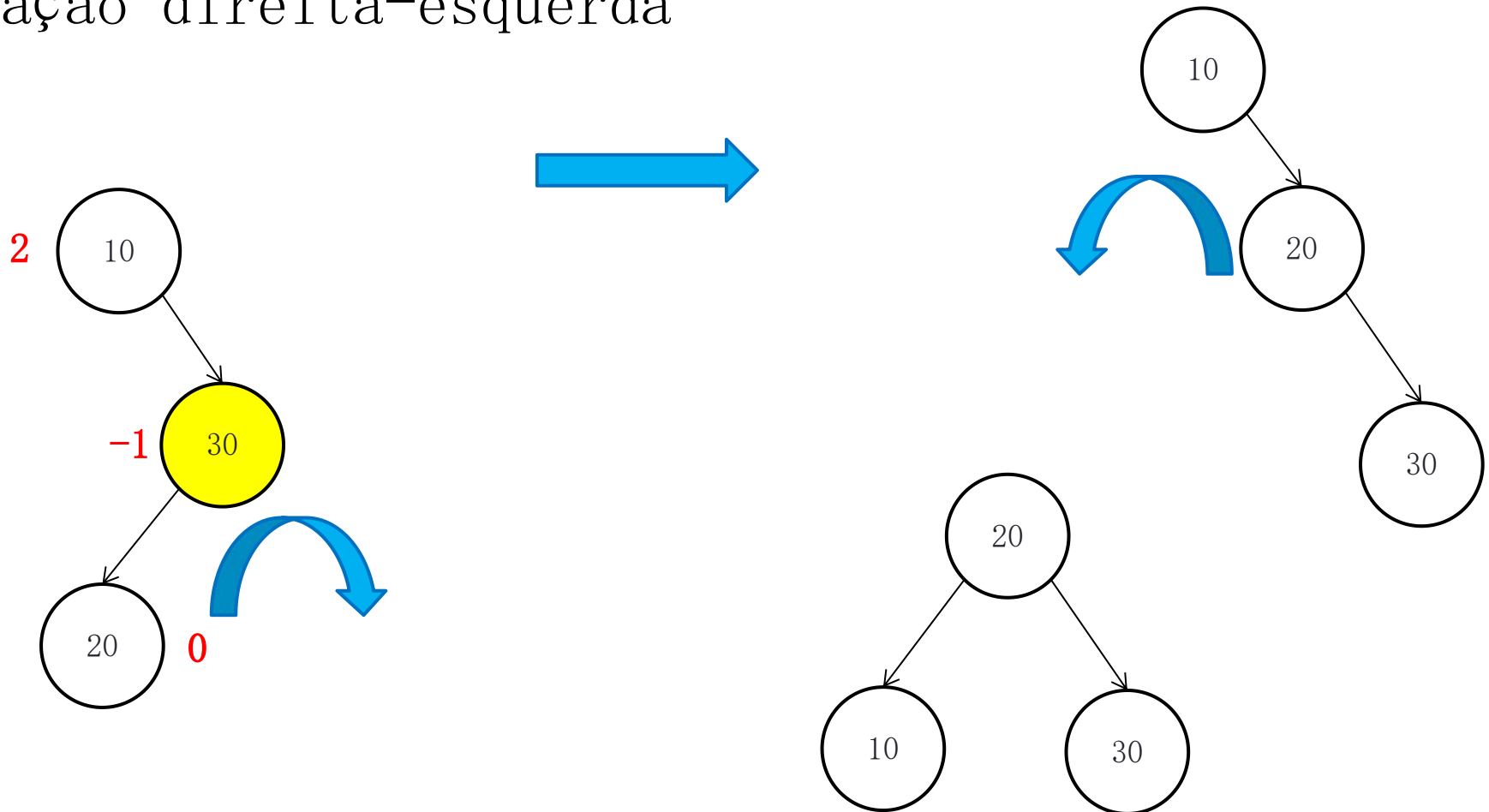
Árvores Balanceadas: AVL

□ Rotação esquerda-direita



Árvores Balanceadas: AVL

□ Rotação direita-esquerda

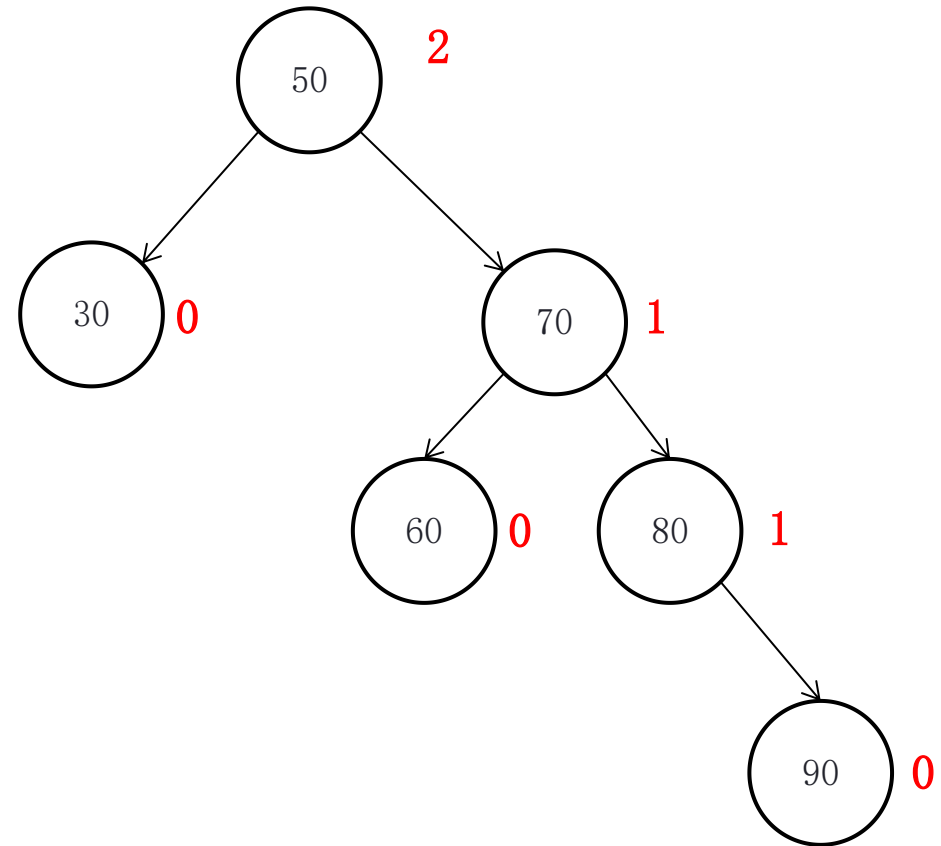


Árvores Balanceadas: AVL

Calcular fator de balanceamento (FB).

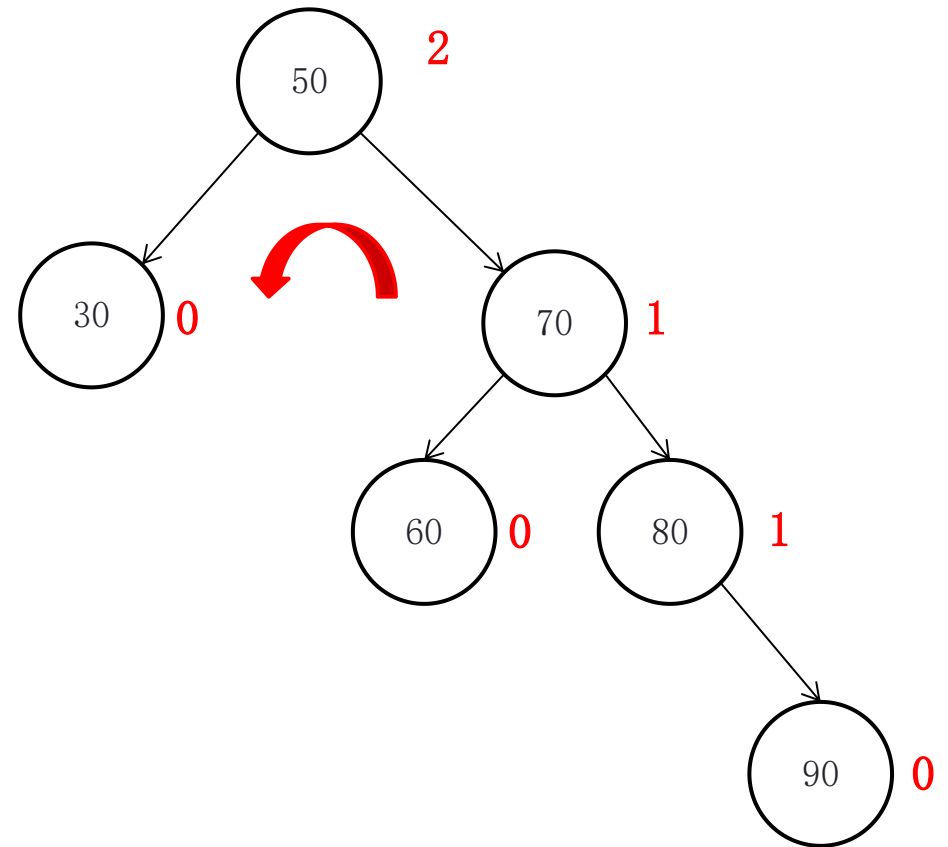
Se FB for diferente de 1, 0 ou -1:

- Se FB é negativo, as rotações são feitas à direita.
- Se FB é positivo, as rotações são feitas à esquerda.



Árvores Balanceadas: AVL

Fator de balanceamento positivo.



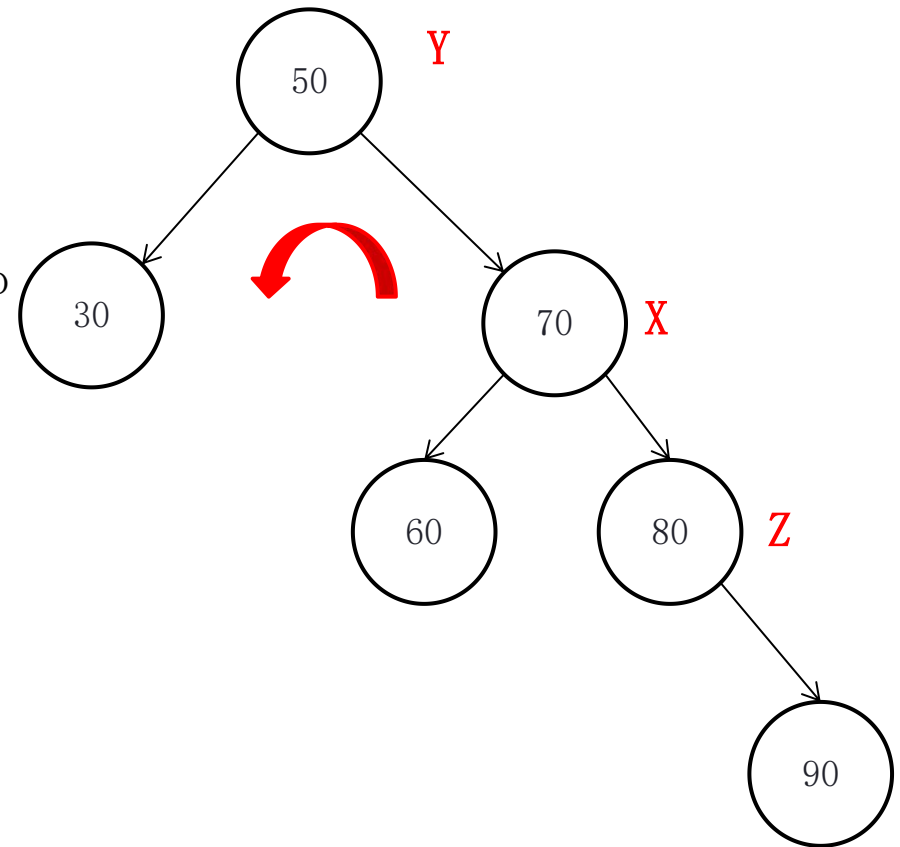
Árvores Balanceadas: AVL

O nó X toma o lugar do nó Y (fator 2)

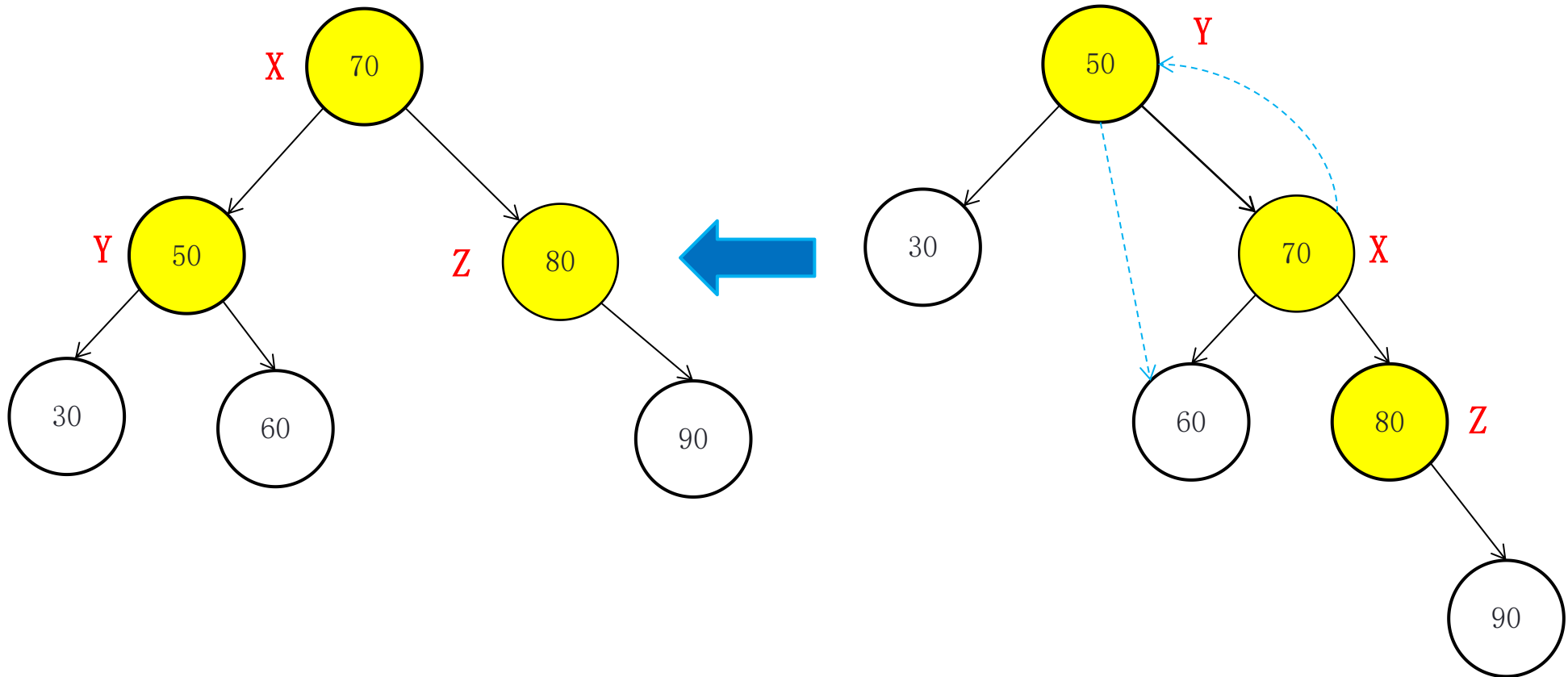
A sub-árvore direita do nó X permanece

A sub-árvore esquerda do nó X será colocada como sub-árvore direita do nó Y

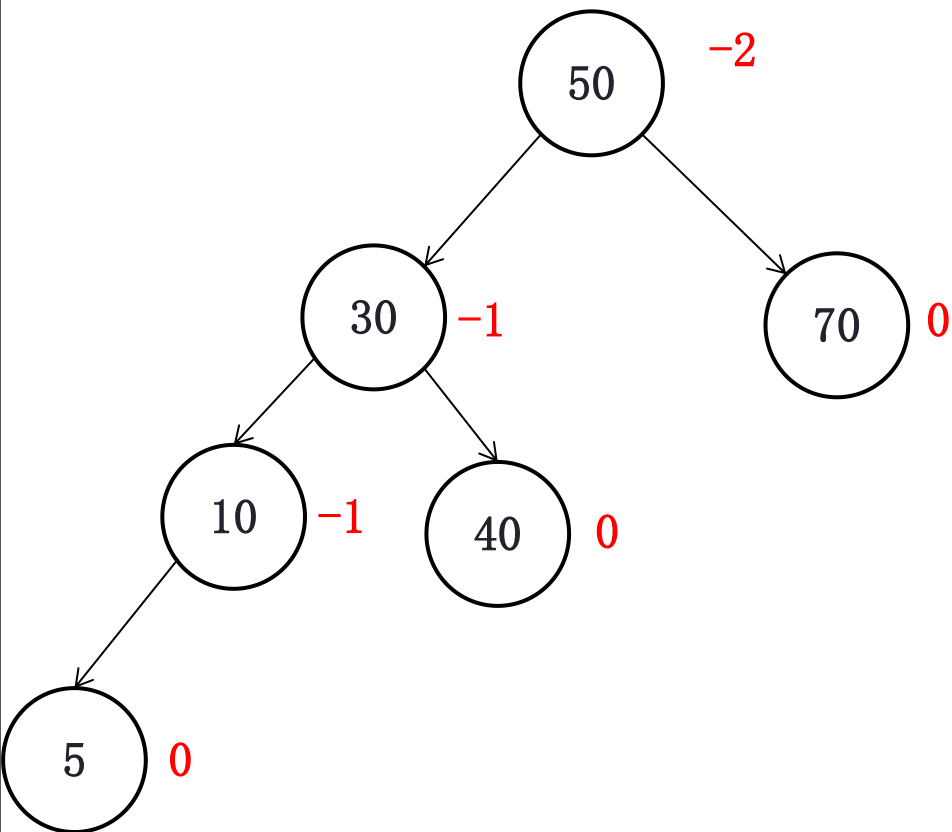
O filho esquerdo do nó X aponta para o nó Y



Árvores Balanceadas: AVL



Árvores Balanceadas: AVL



Calcular fator de balanceamento (FB).

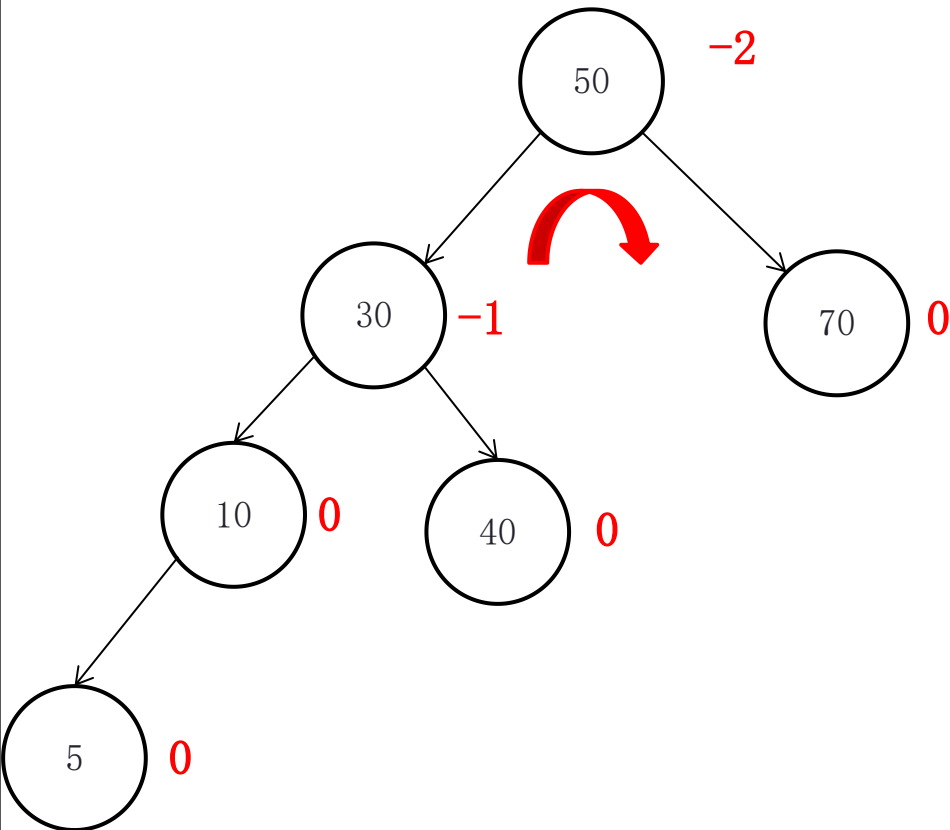
Se FB for diferente de 1, 0 ou -1:

- Se FB é negativo, as rotações são feitas à direita.

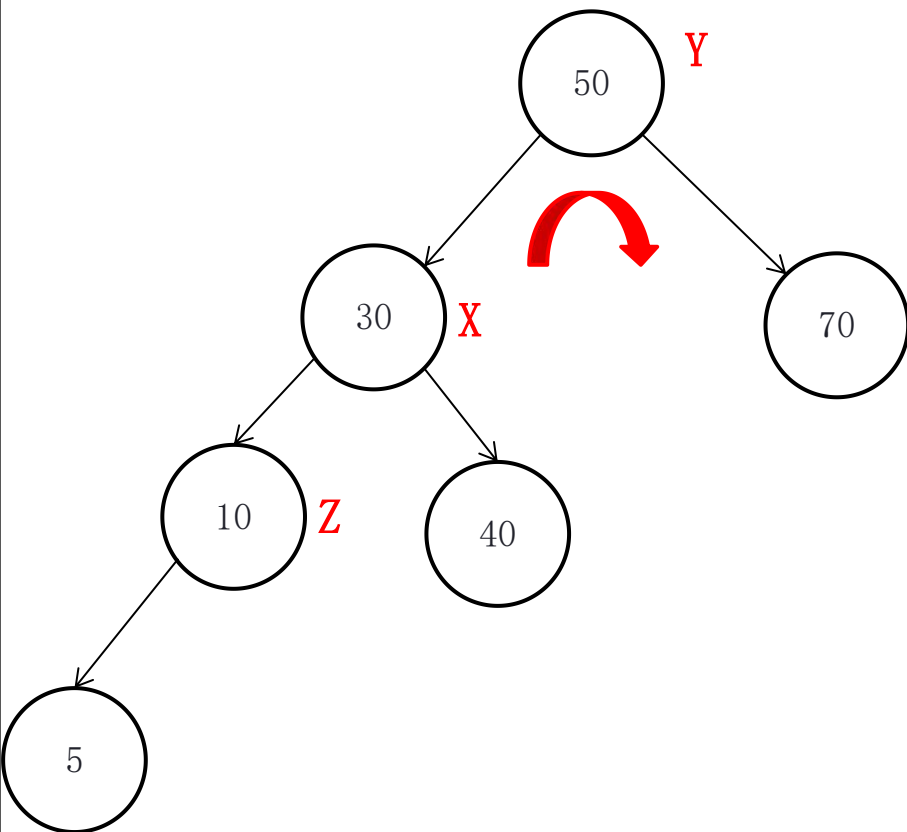
- Se FB é positivo, as rotações são feitas à esquerda.

Árvores Balanceadas: AVL

Fator de balanceamento negativo!



Árvores Balanceadas: AVL



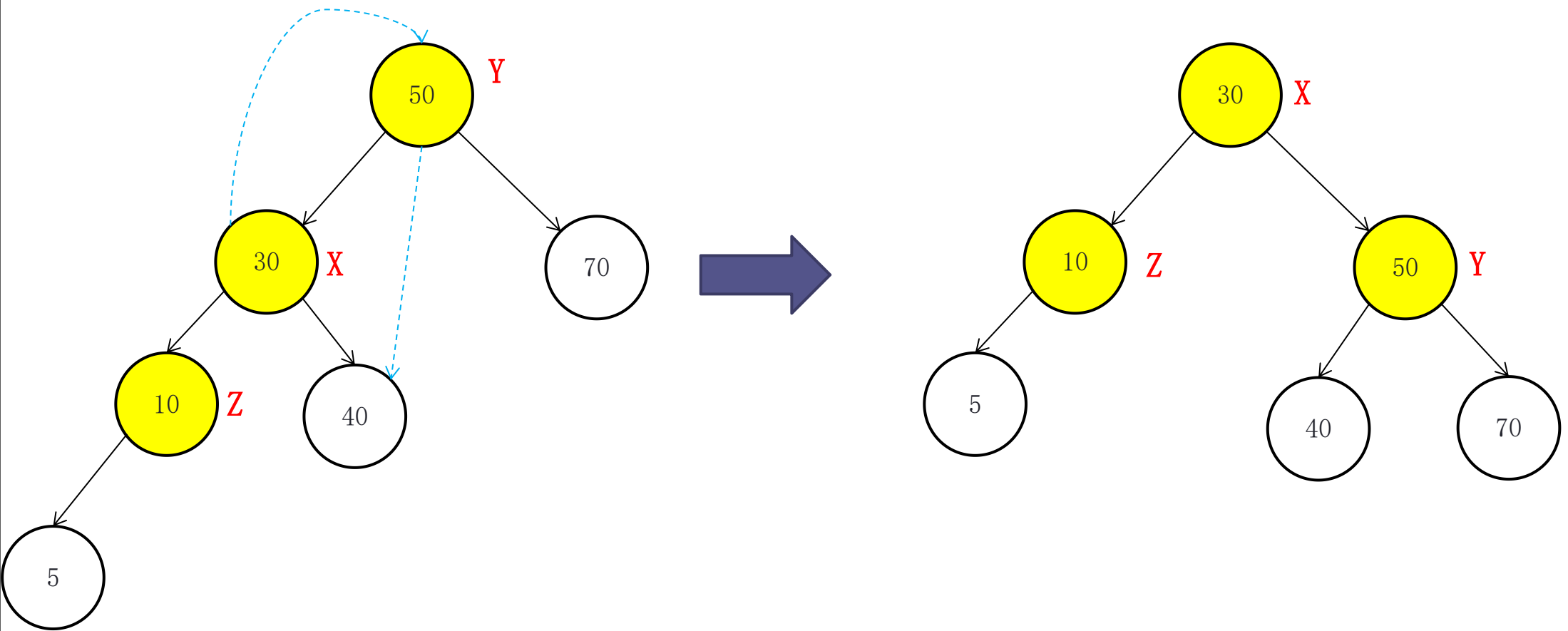
O nó X que está no nível do meio dos três envolvidos toma o lugar do nó com $FB=-2$ (Y)

A sub-árvore esquerda do nó X permanece

A sub-árvore direita do nó X será colocada como sub-árvore esquerda do nó Y

O filho direito do nó X aponta para o nó Y

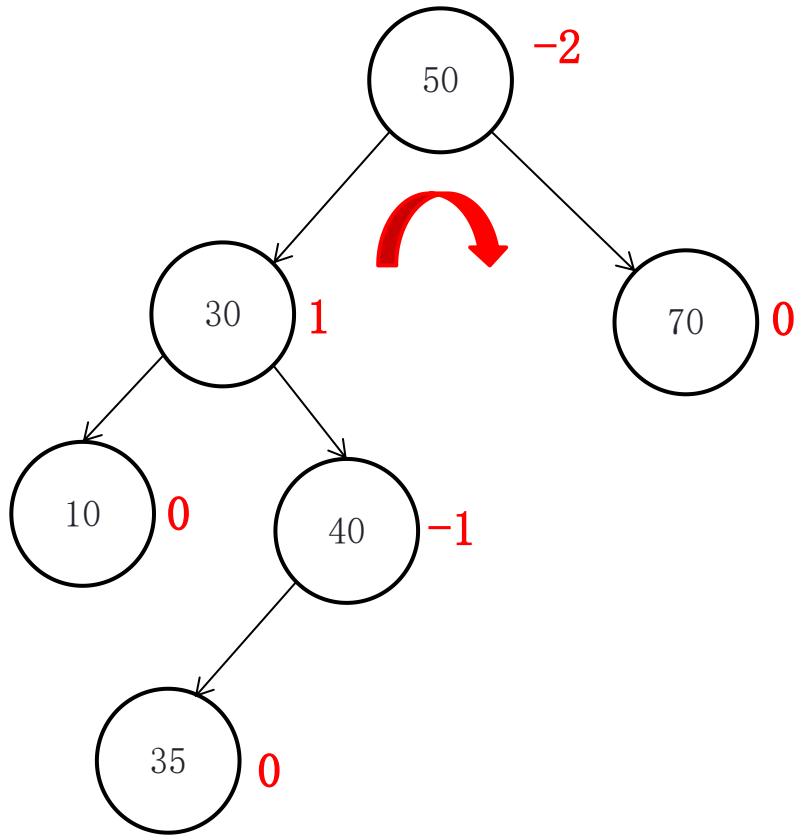
Árvores Balanceadas: AVL



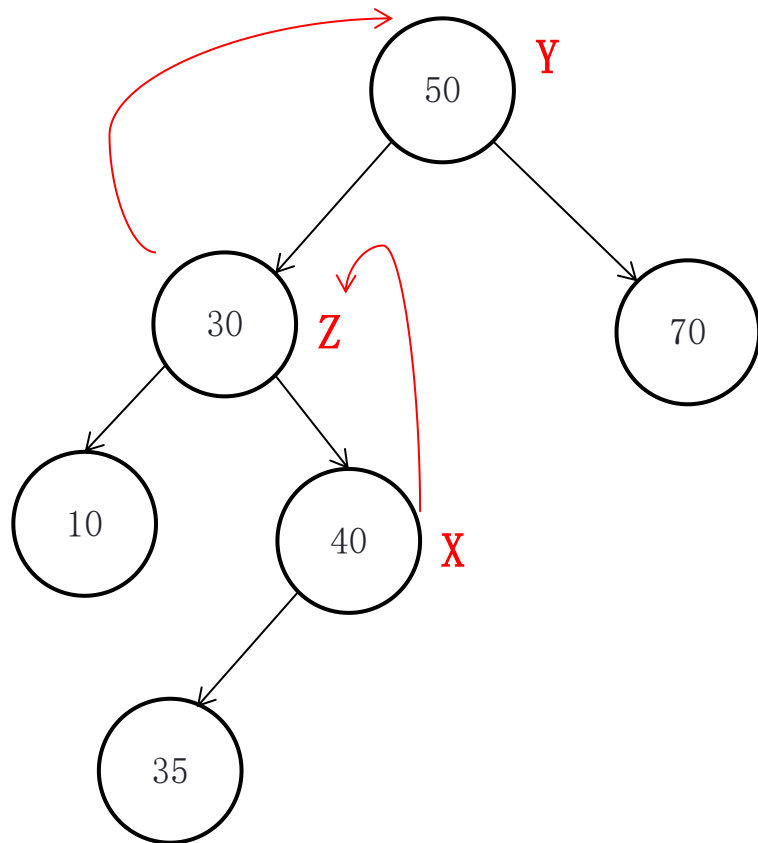
Árvores Balanceadas: AVL

Rotação dupla esquerda-direita

Fator de balanceamento negativo.



Árvores Balanceadas: AVL



O nó que está no nível mais alto das três envolvidas (nó X) toma o lugar da célula cujo fator de balanceamento é +2 (nó Y)

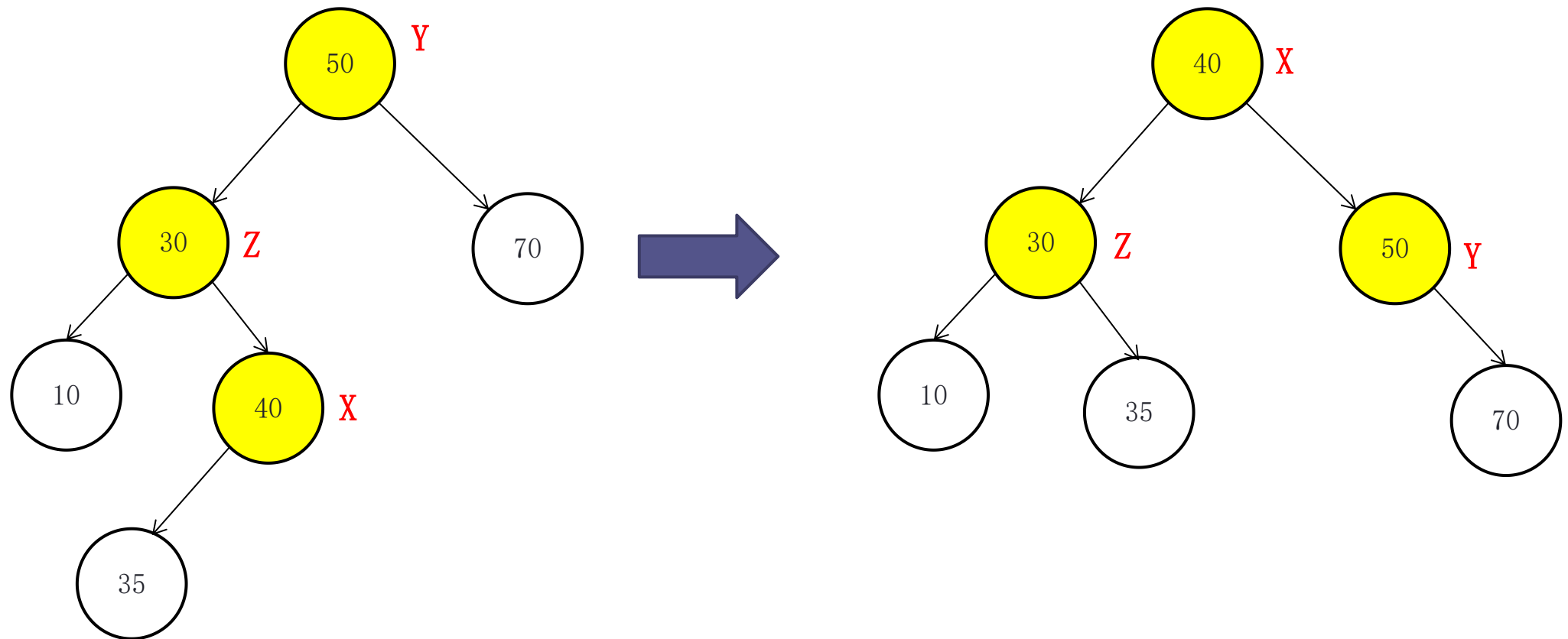
A sub-árvore direita do nó X será colocada como sub-árvore esquerda do nó Y

A sub-árvore esquerda do nó X será colocada como sub-árvore direita do nó Z

O filho direito do nó X aponta para o nó Y

O filho esquerdo do nó X aponta para o nó Z

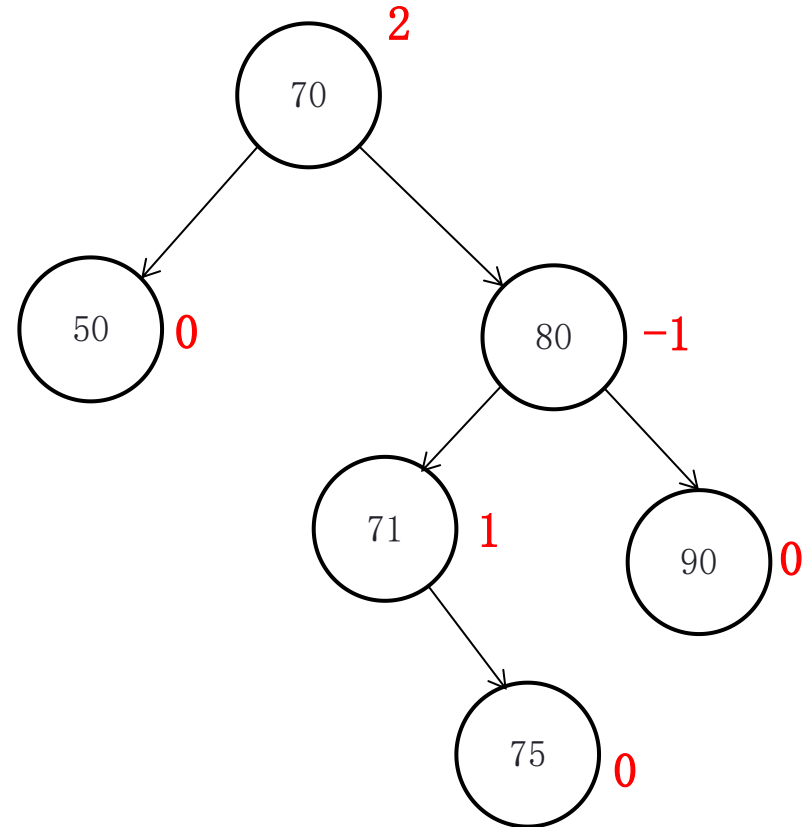
Árvores Balanceadas: AVL



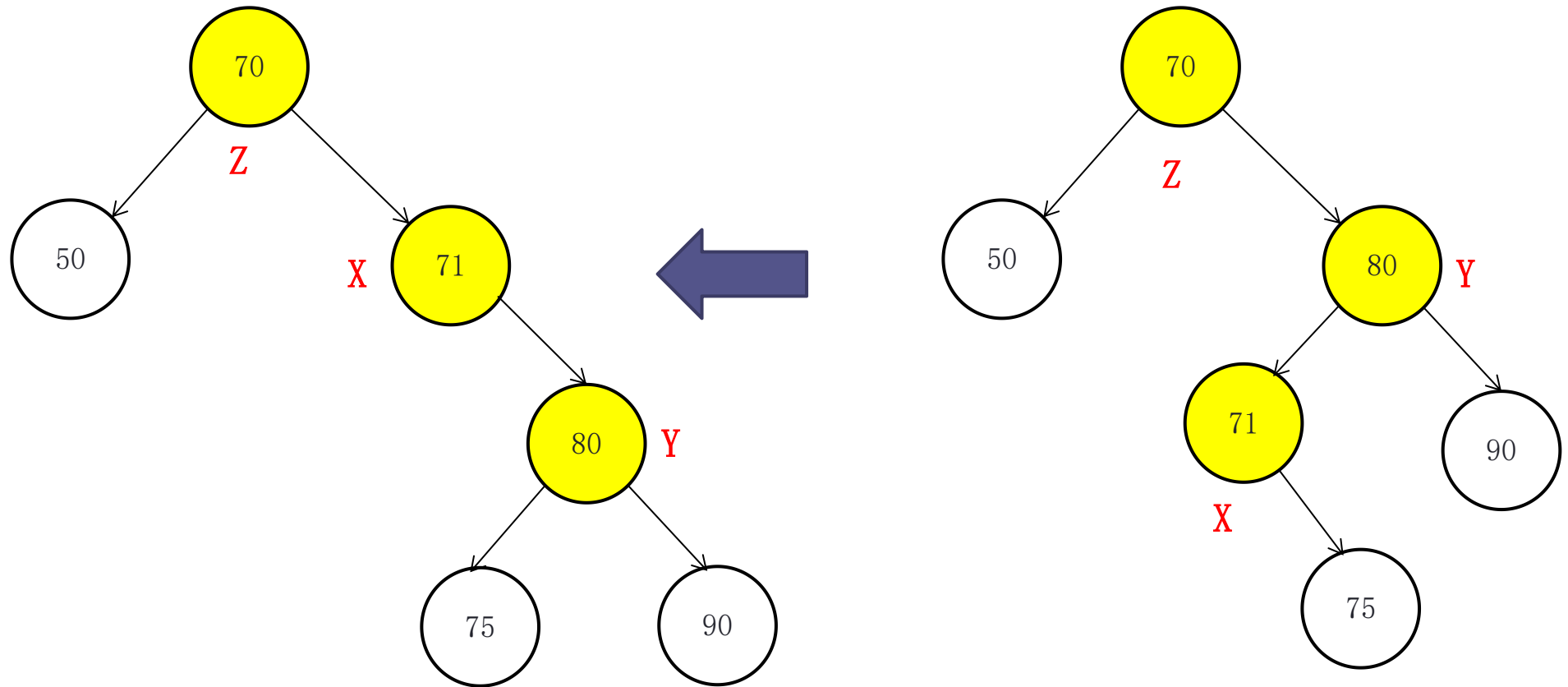
Árvores Balanceadas: AVL

Rotação dupla direita-esquerda

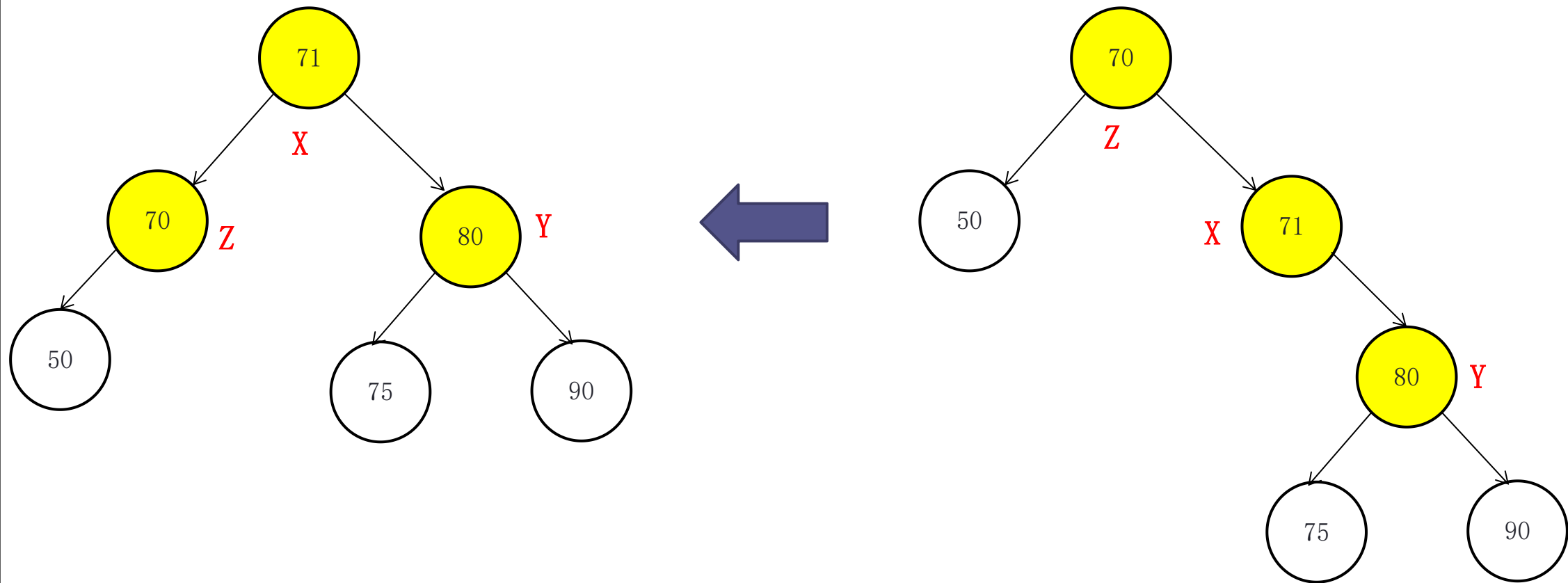
Fator de balanceamento positivo.



Árvores Balanceadas: AVL



Árvores Balanceadas: AVL



Árvores Balanceadas: AVL

□ Inserção em árvore AVL

Para inserir um valor **V** na árvore:

- Se raiz é **NULL** : insira o nó
- **V** é **menor** do que a raiz: vá para sub-árvore **esquerda**
- **V** é **maior** do que a raiz: vá para sub-árvore **direita**
- *Aplique o método recursivamente*
- Ao voltar na recursão, recalcule as alturas de cada sub-árvore
- Aplique a **rotação** necessária se o FB for **+2** ou **-2**

Árvores Balanceadas: AVL

❑ Remoção em árvore AVL

Para remover um valor **V** na árvore:

- Se raiz é **NULL** : erro
- **V** é **menor** do que a raiz: vá para sub-árvore **esquerda**
- Exclua o valor
- Recalcula altura e FB : rotaciona
- **V** é **maior** do que a raiz: vá para sub-árvore **direita**
- Exclua o valor
- Recalcula altura e FB : rotaciona

Árvores Balanceadas: Splay

- ❑ É um tipo de árvore binária de pesquisa.
- ❑ Também chamadas de árvores auto-ajustadas ou árvore de afunilamento.
- ❑ São mais simples que AVL:
 - ❑ não forçam o equilíbrio.
 - ❑ Não mantêm informação de altura.

Árvores Balanceadas: Splay

- ❑ Objetivo: tornar mais acessível o que é mais usado.
- ❑ Junto a raiz estão os elementos:
 - ❑ mais usados.
 - ❑ Mais recentes.
- ❑ Cada vez que um nó é pesquisado, ele é puxado para raiz.

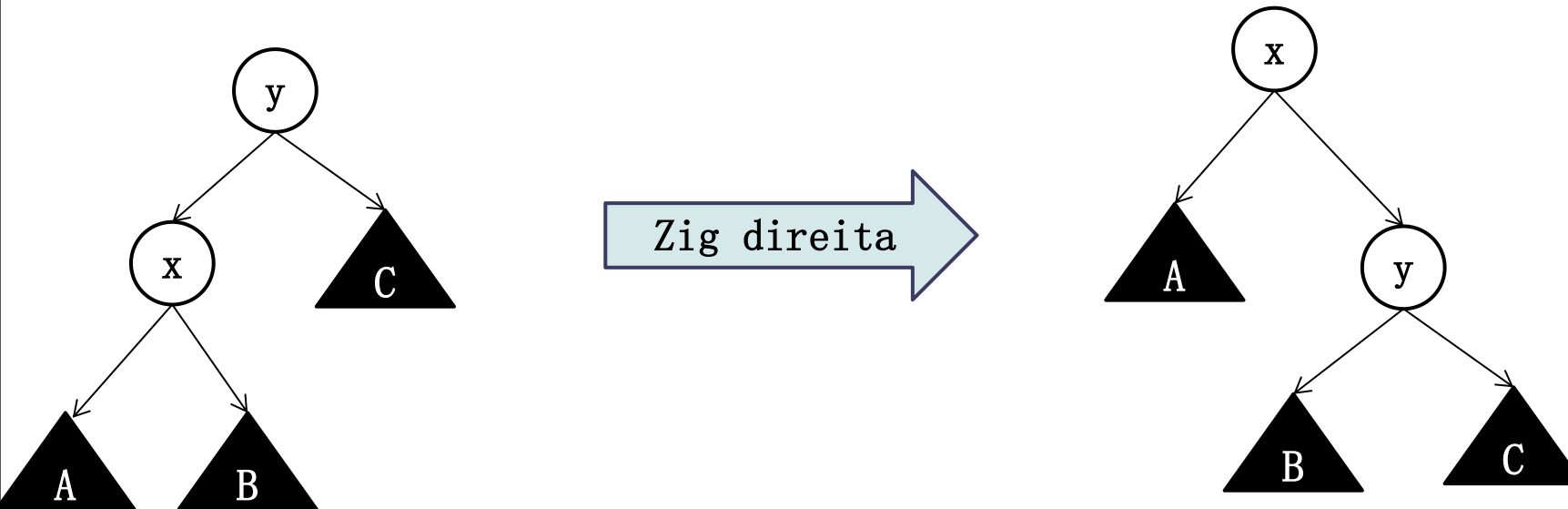
Árvores Balanceadas: Splay

- ❑ Fazer **splay** é trazer um elemento **x** para a raiz da árvore utilizando sucessivas rotações e tantas quanto necessárias.
- ❑ Operações: pesquisa, inserção e remoção.
- ❑ Em todas as operações, a árvore faz splay.

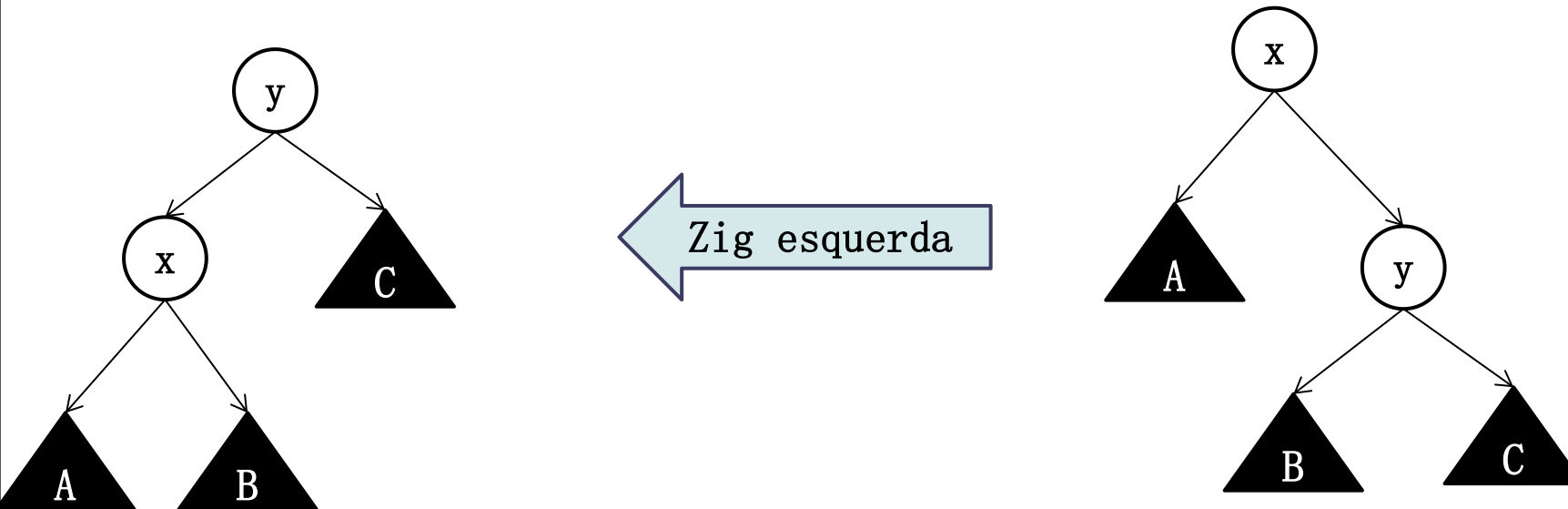
Árvores Balanceadas: Splay

- ❑ Rotações simples:
 - ❑ zig direita.
 - ❑ zig esquerda
- ❑ Funciona da mesma forma que as rotações simples da árvore AVL.

Árvores Balanceadas: Splay

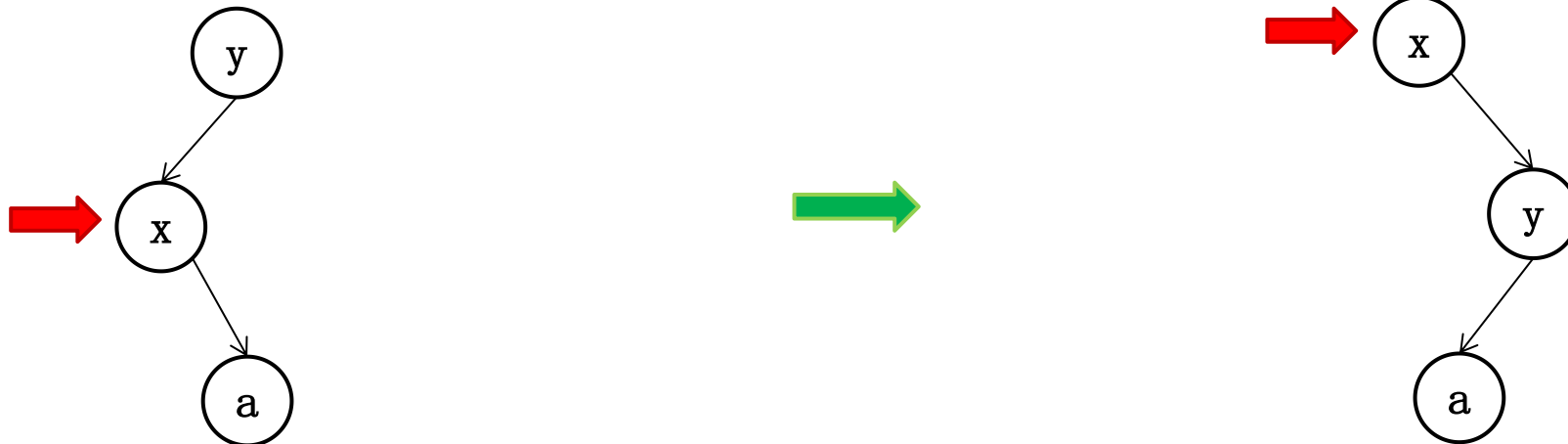


Árvores Balanceadas: Splay



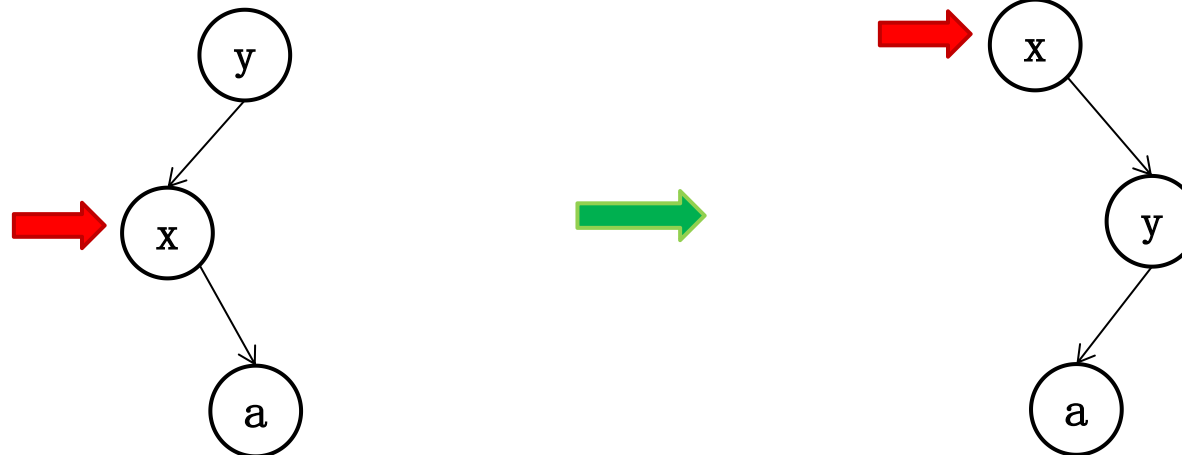
Rotação Zig (à direita)

□ Ocorre quando o nó a ser movido para a raiz é o filho esquerdo da raiz.



Rotação Zig (à direita)

1. x é o filho esquerdo da raiz
2. a subárvore direita de x se torna filho esquerdo de y
3. y se torna filho direito de x
4. x é a nova raiz após a rotação

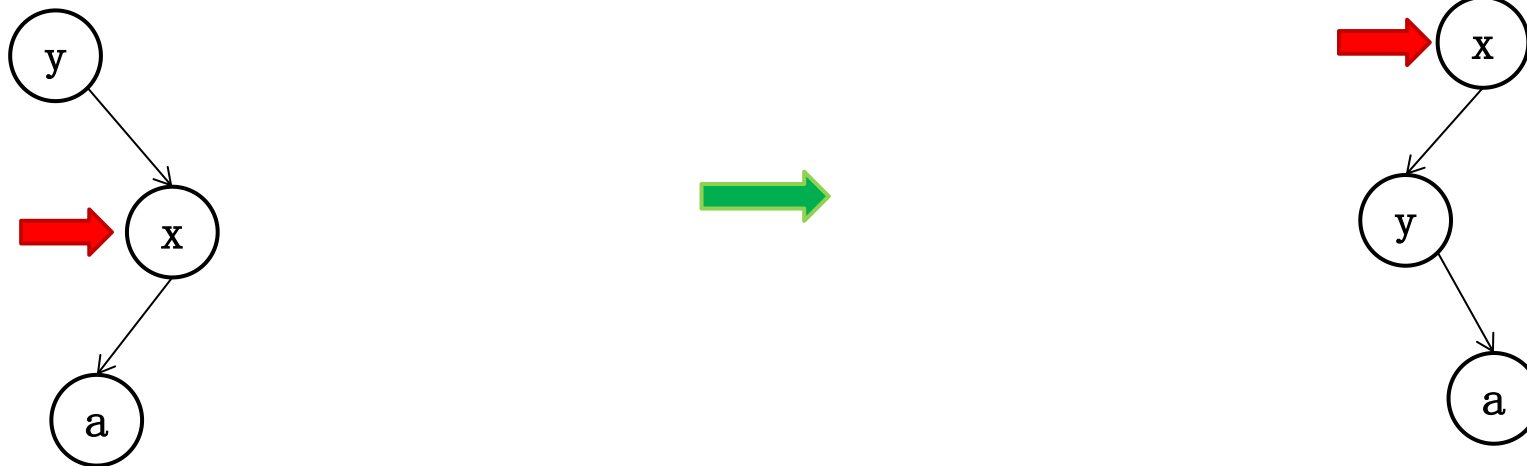


Rotação Zig (à direita)

```
FUNÇÃO rotacaoDireita(y)
  x = y.esquerdo
  y.esquerdo = x.direito
  x.direito = y
  RETORNE x
FIM FUNÇÃO
```

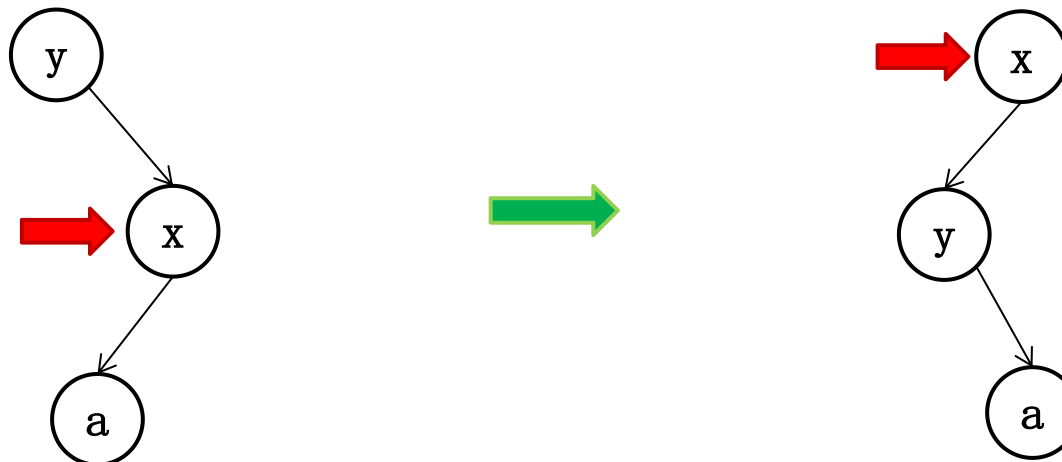
Rotação Zig (à esquerda)

- Ocorre quando o nó a ser movido para a raiz é o filho direito da raiz.



Rotação Zig (à esquerda)

1. x é o filho direito da raiz
2. a subárvore esquerda de x se torna filho direito de y
3. y se torna filho esquerdo de x
4. x é a nova raiz após a rotação



Rotação Zig (à esquerda)

```
FUNÇÃO rotacaoEsquerda(y)
  x = y.direito
  y.direito = x.esquerdo
  x.esquerdo = y
  RETORNE x
FIM FUNÇÃO
```

Árvores Balanceadas: Splay

□ Rotações dupla:

□ zig zig.

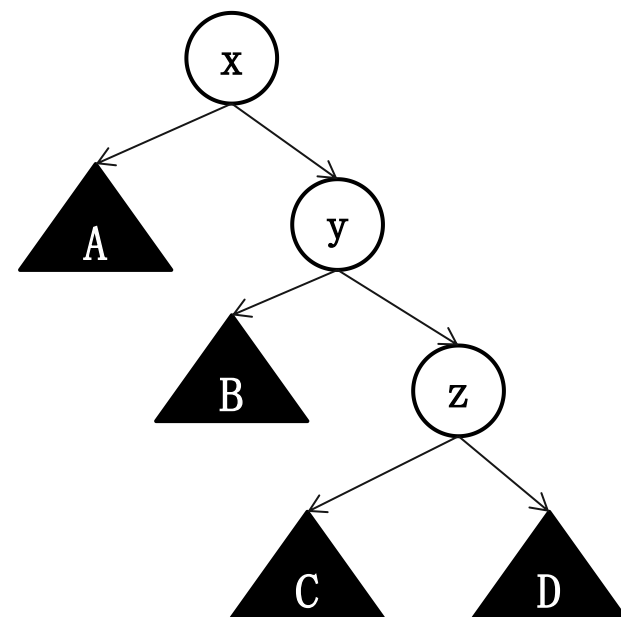
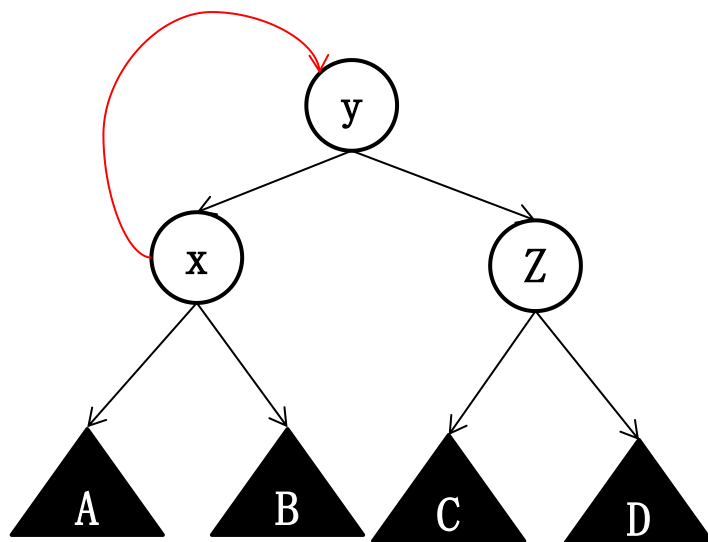
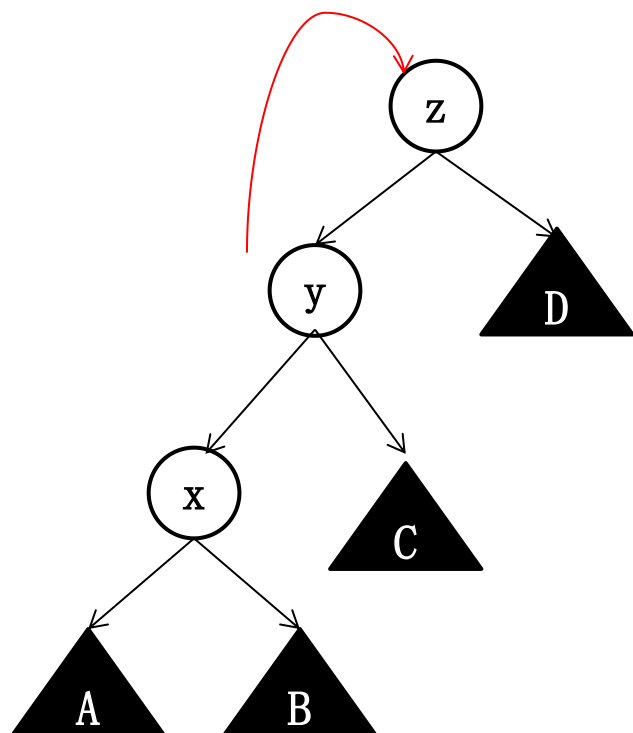
□ zig zag.

Árvores Balanceadas: Splay

□ Zig zig

□ faz zig de pai de X.

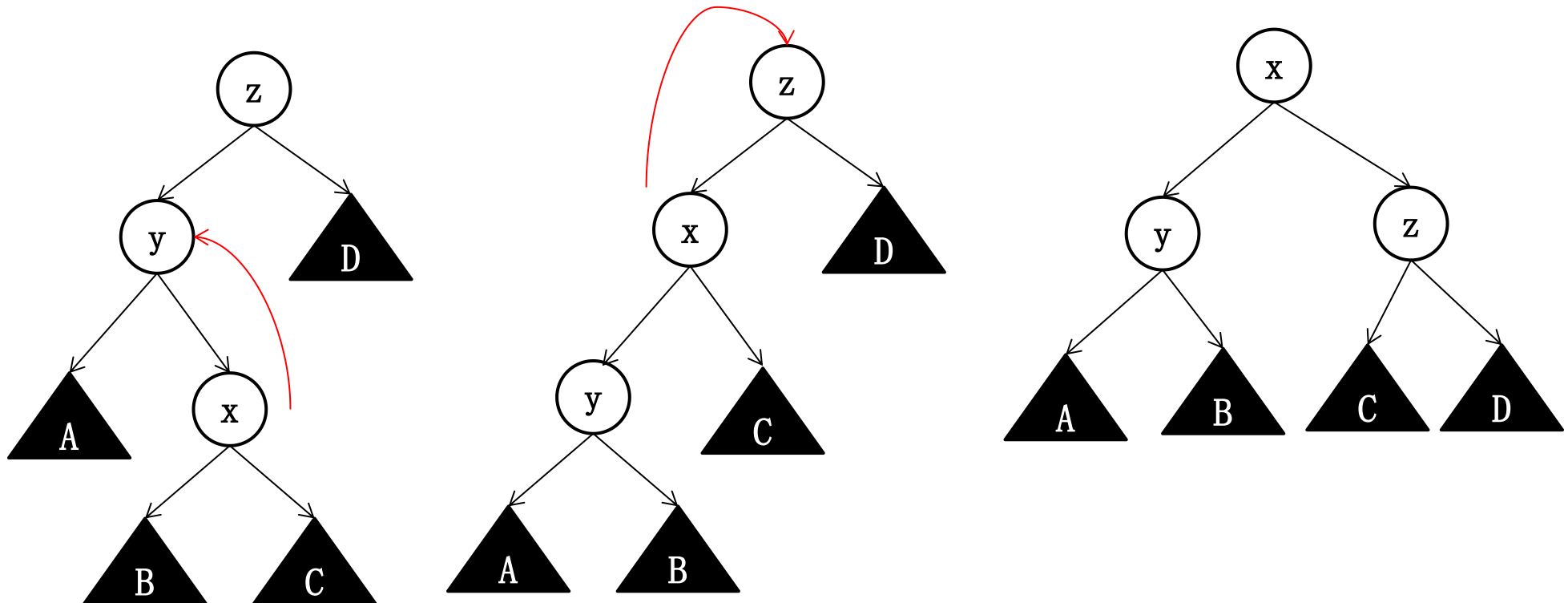
□ faz zig de X.



Árvores Balanceadas: Splay

□ Zig zag

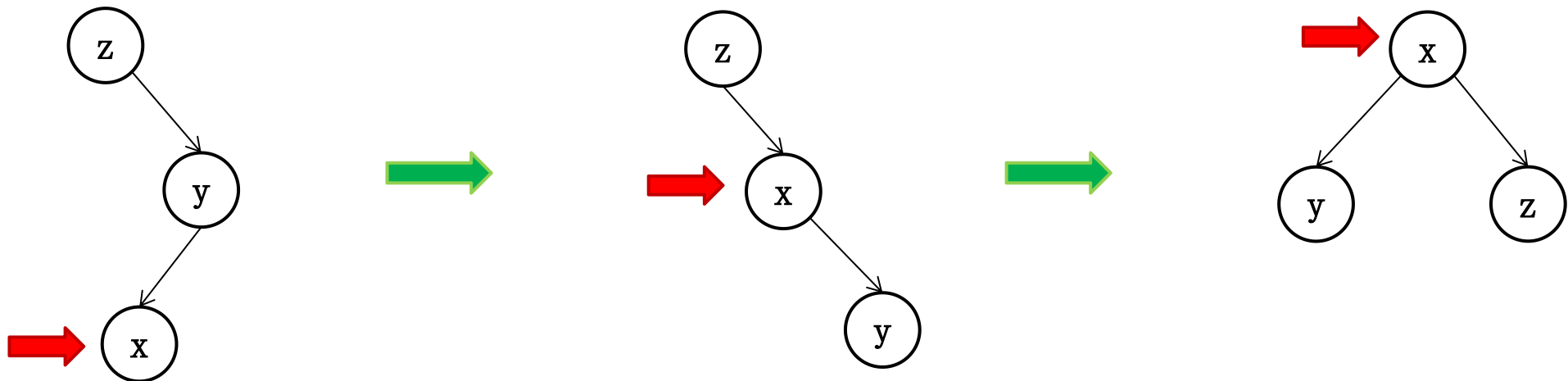
- faz zig de X com o pai de X.
- faz zig de X com o avô de X.



Rotação zig zag (esquerda-direta)

□ O nó X é o filho esquerdo de Y , e Y é o filho direito de Z . Para corrigir isso, faz duas rotações consecutivas:

- Rotação à direita em Y (transformando X no pai de Y).
- Rotação à esquerda em Z (transformando X na nova raiz).



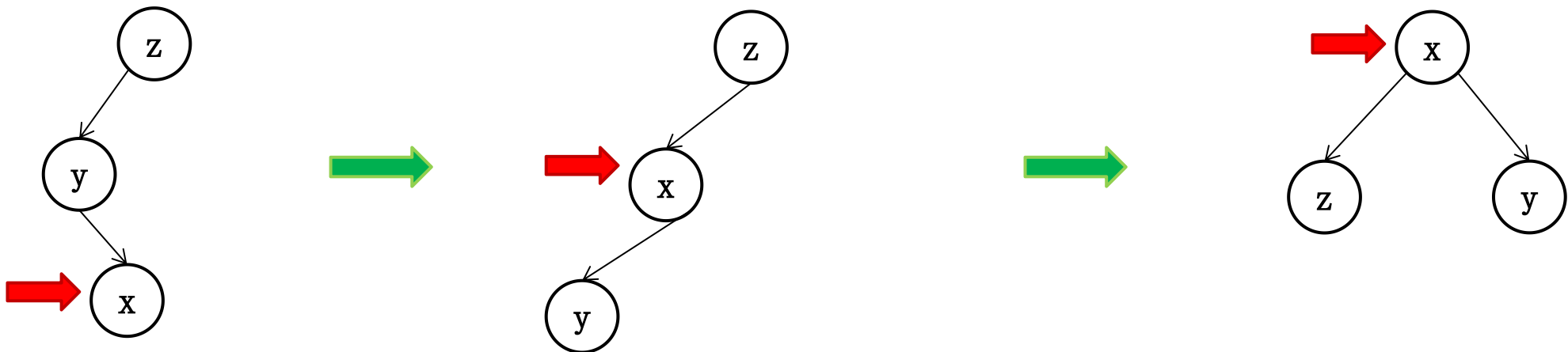
Rotação zig zag (esquerda-direta)

```
FUNÇÃO zigZagEsquerdaDireita(z)
    z.esquerdo = rotacaoEsquerda(z.esquerdo)
    RETORNE rotacaoDireita(z)
FIM FUNÇÃO
```

Rotação zig zag (direita-esquerda)

□ O nó X é o filho direito de Y , e Y é o filho esquerdo de Z . Para corrigir isso, faz duas rotações consecutivas:

- Rotação à esquerda em Y (transformando X no pai de Y).
- Rotação à direita em Z (transformando X na nova raiz).



Rotação zig zag (direita-esquerda)

```
FUNÇÃO zigZagDireitaEsquerda(z)  
    z.direito = rotacaoDireita(z.direito)  
    RETORNE rotacaoEsquerda(z)  
FIM FUNÇÃO
```

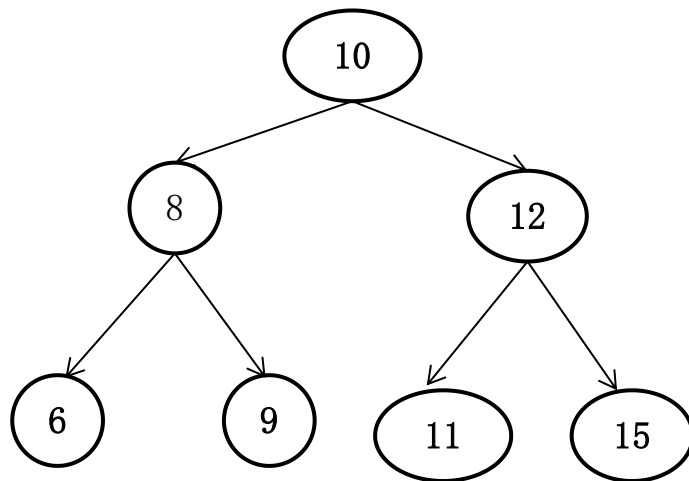
Árvores Balanceadas: Splay

□ Pesquisa (x, A)

Faz percurso até encontrar x

Se existe faz SPLAY de x

Senão faz SPLAY do sucessor mais próximo de x



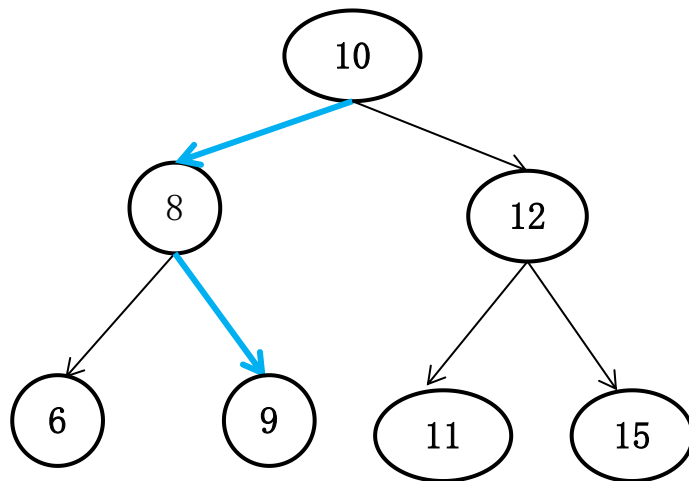
Árvores Balanceadas: Splay

□ Pesquisa (x, A)

Faz percurso até encontrar x

Se existe faz SPLAY de x

Senão faz SPLAY do sucessor mais próximo de x



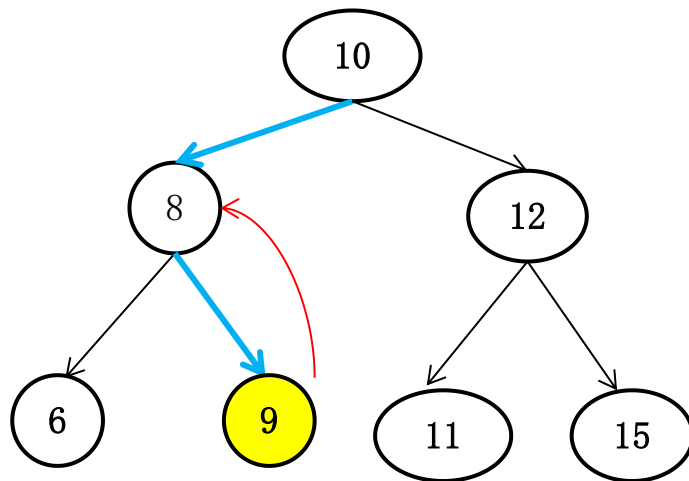
Árvores Balanceadas: Splay

□ Pesquisa (x, A)

Faz percurso até encontrar x

Se existe faz SPLAY de x

Senão faz SPLAY do sucessor mais próximo de x



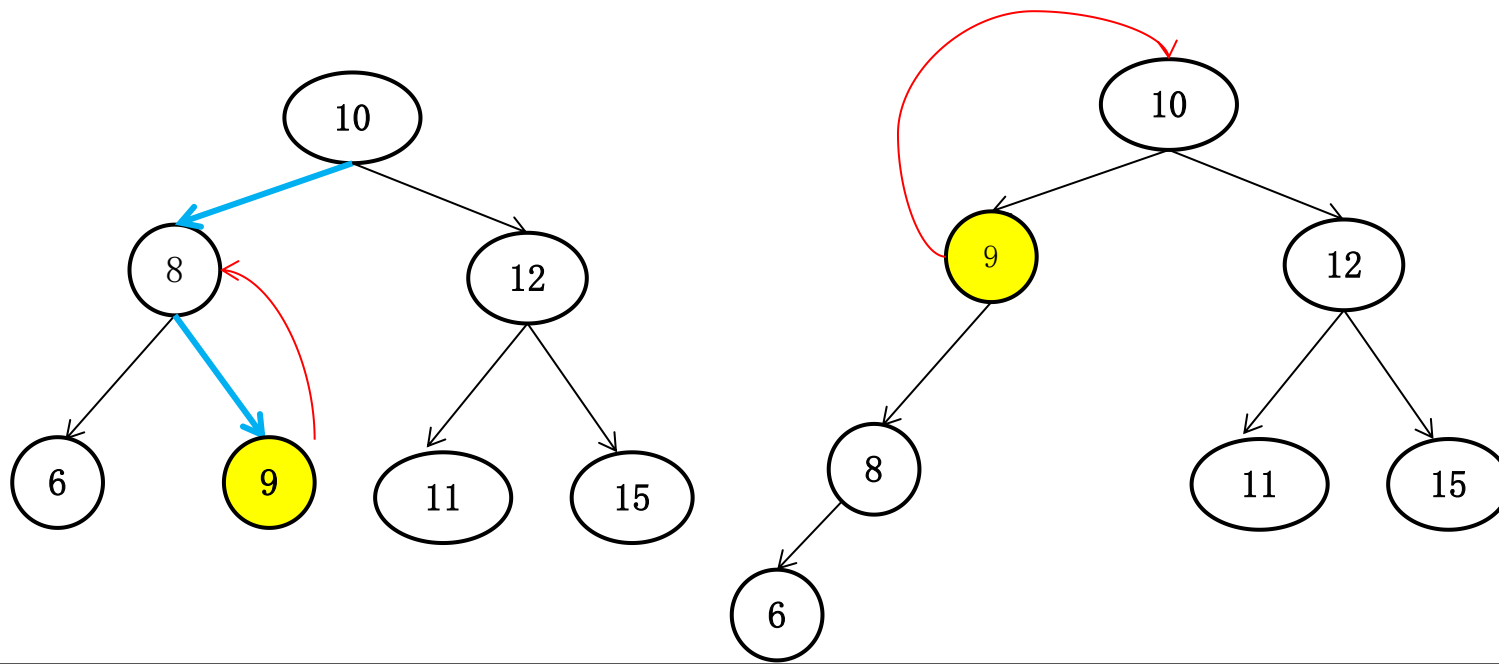
Árvores Balanceadas: Splay

□ Pesquisa (x, A)

Faz percurso até encontrar x

Se existe faz SPLAY de x

Senão faz SPLAY do sucessor mais próximo de x



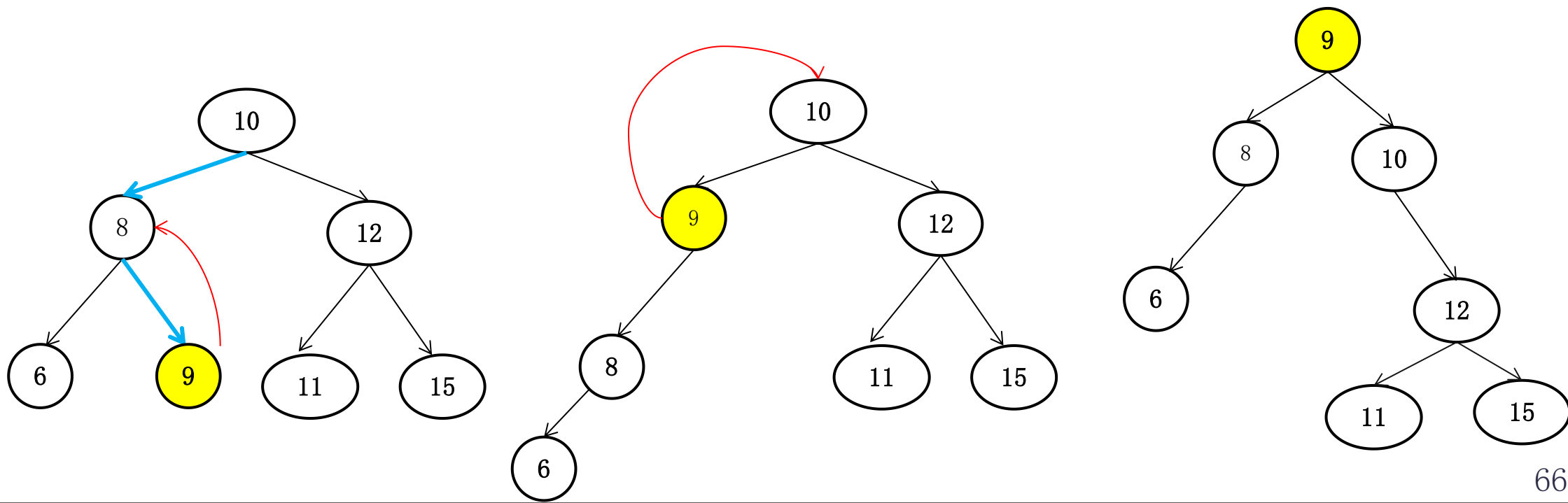
Árvores Balanceadas: Splay

□ Pesquisa (x, A)

Faz percurso até encontrar x

Se existe faz SPLAY de x

Senão faz SPLAY do sucessor mais próximo de x



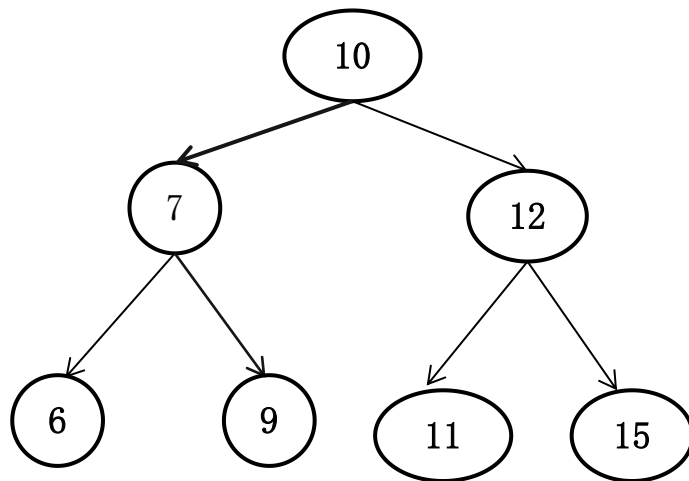
Árvores Balanceadas: Splay

□ Inserção (x, A)

Faz splay de x

Como não existe, o elemento menor mais próximo de x fica na raiz

Insere x



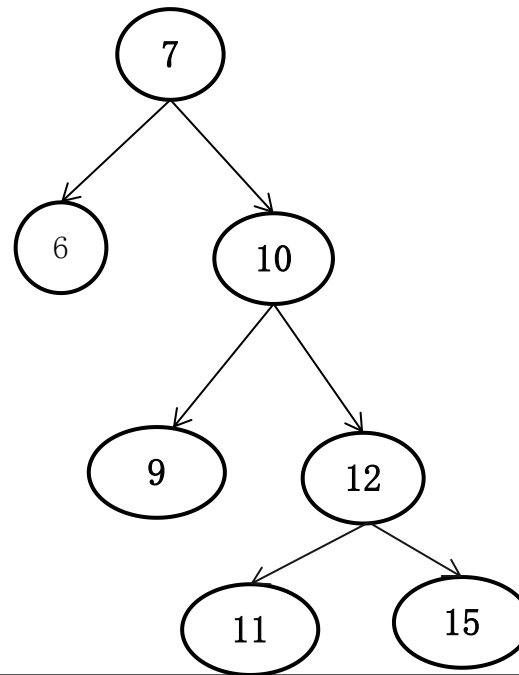
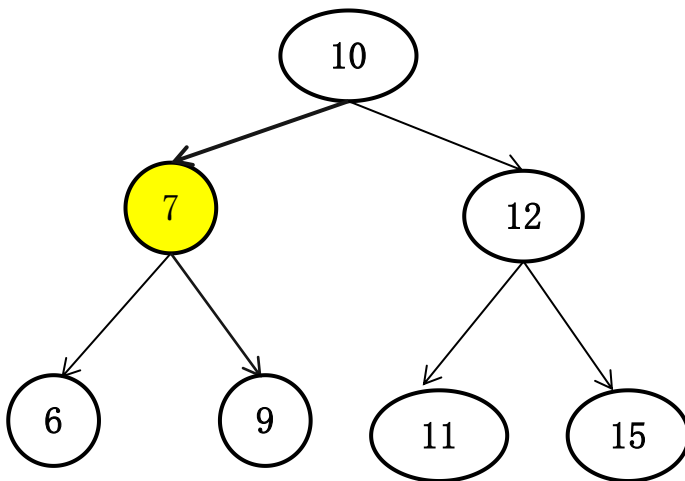
Árvores Balanceadas: Splay

□ Inserção (x, A)

Faz splay de x

Como não existe, o elemento menor mais próximo de x fica na raiz

Insere x



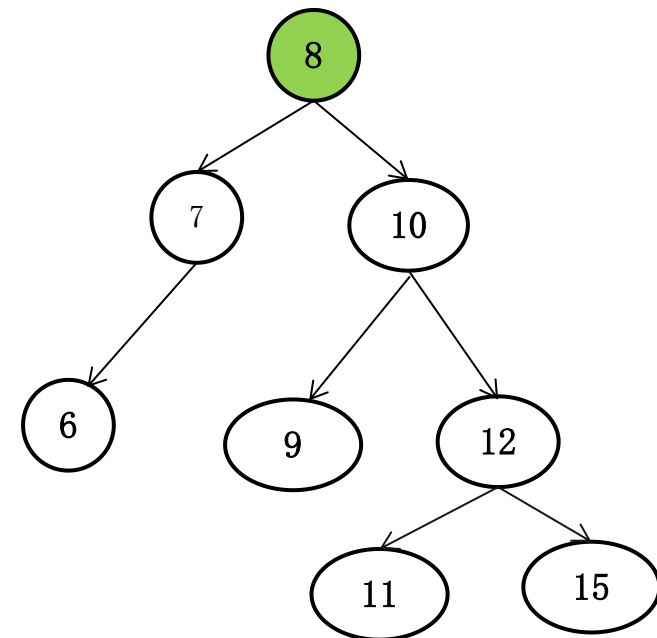
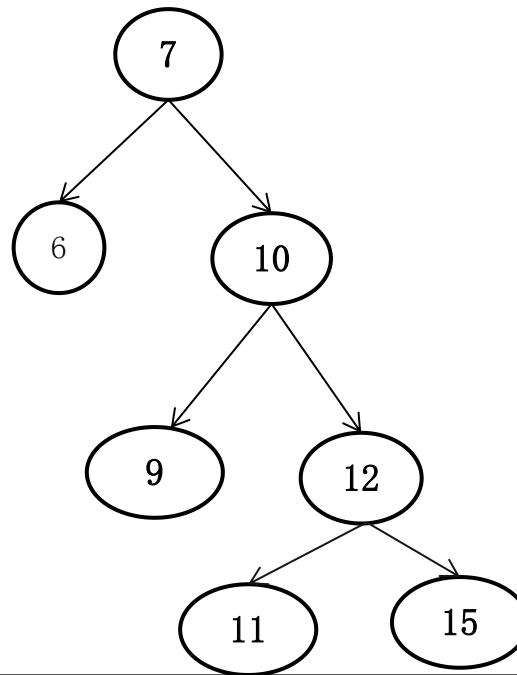
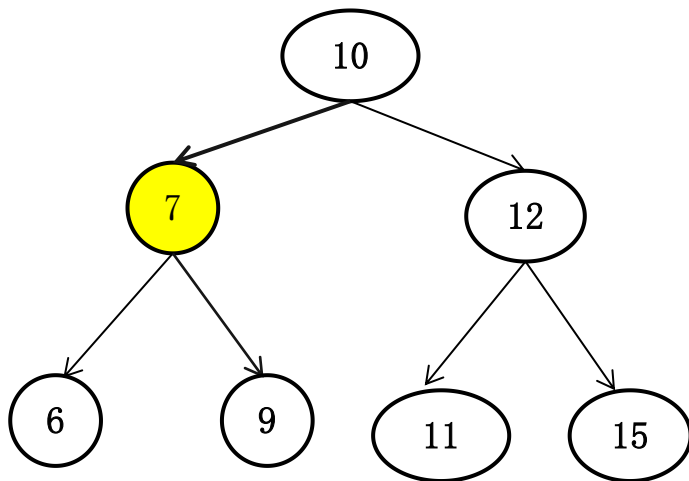
Árvores Balanceadas: Splay

□ Inserção (x, A)

Faz splay de x

Como não existe, o elemento menor mais próximo de x fica na raiz

Insere x



Árvores Balanceadas: Splay

□ Remoção (x, A)

Busca x

Se existe

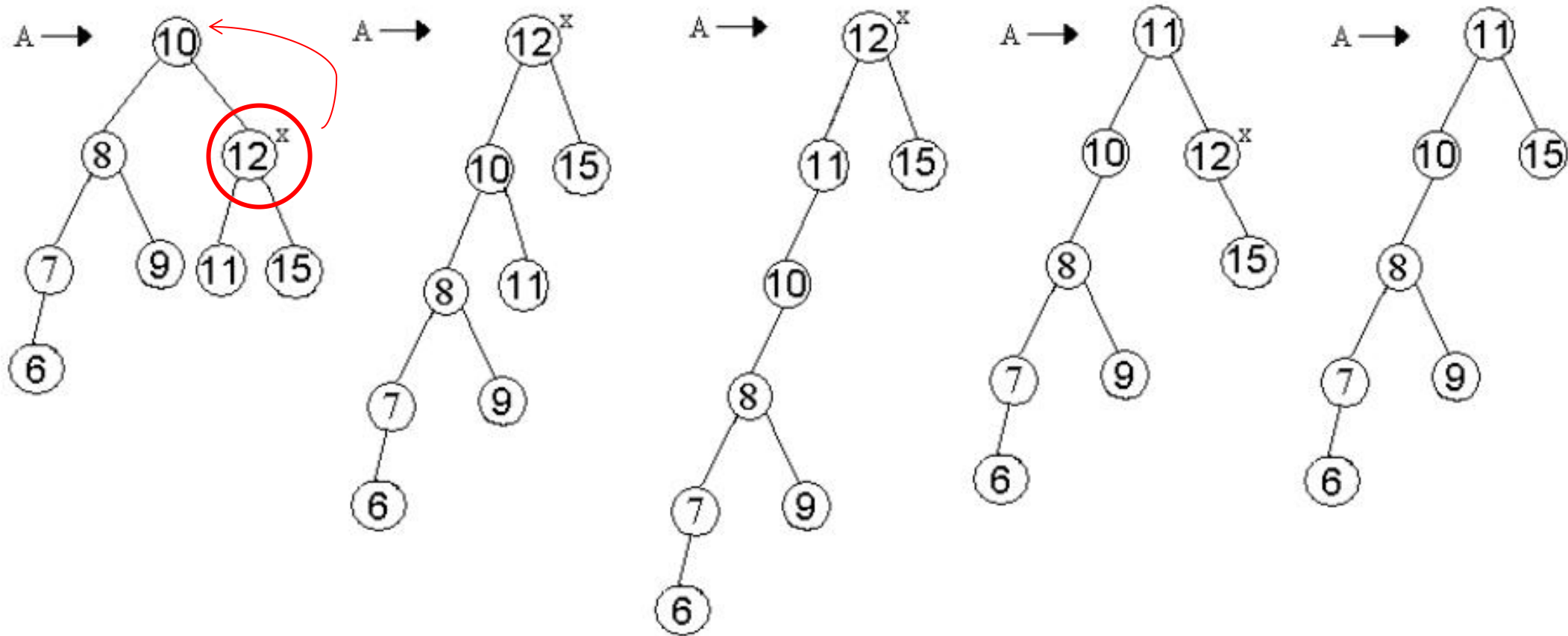
Faz splay de x

Faz splay de x , na sub-árvore esquerda

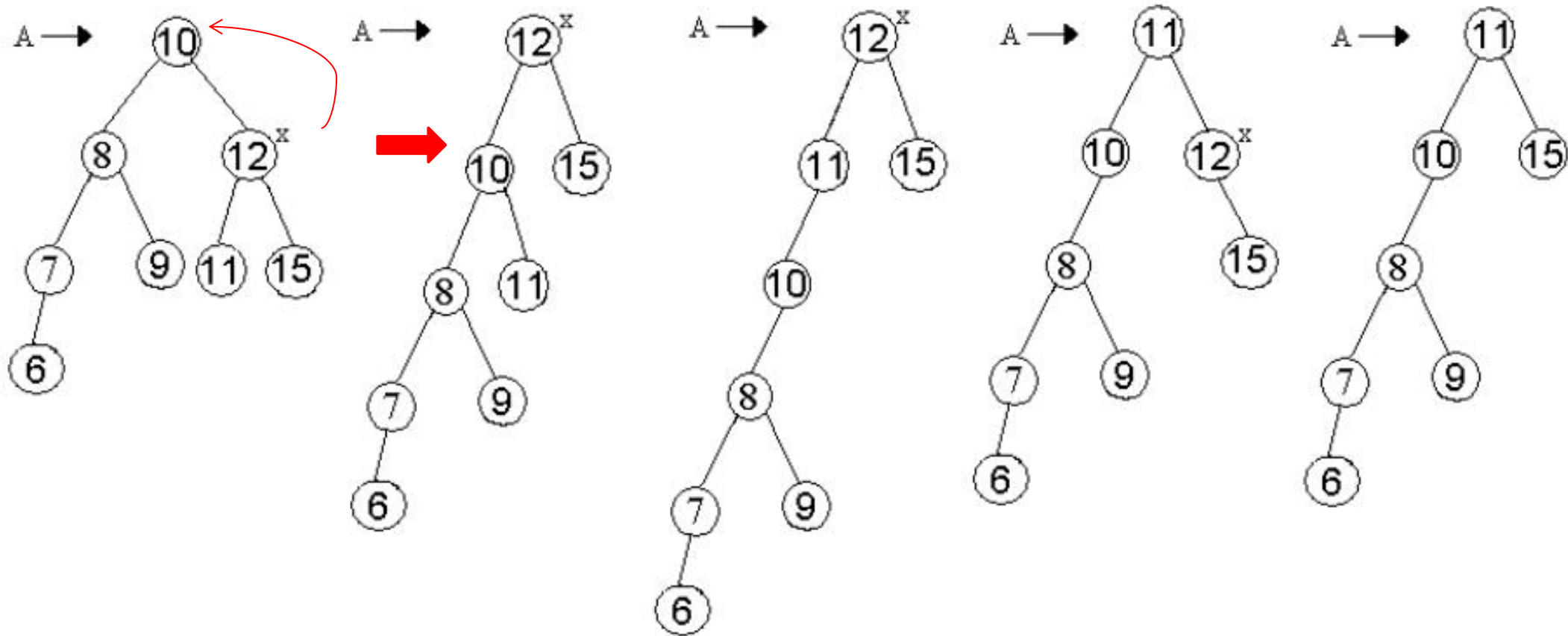
Traz elemento menor mais próximo de x para raiz

Remove x

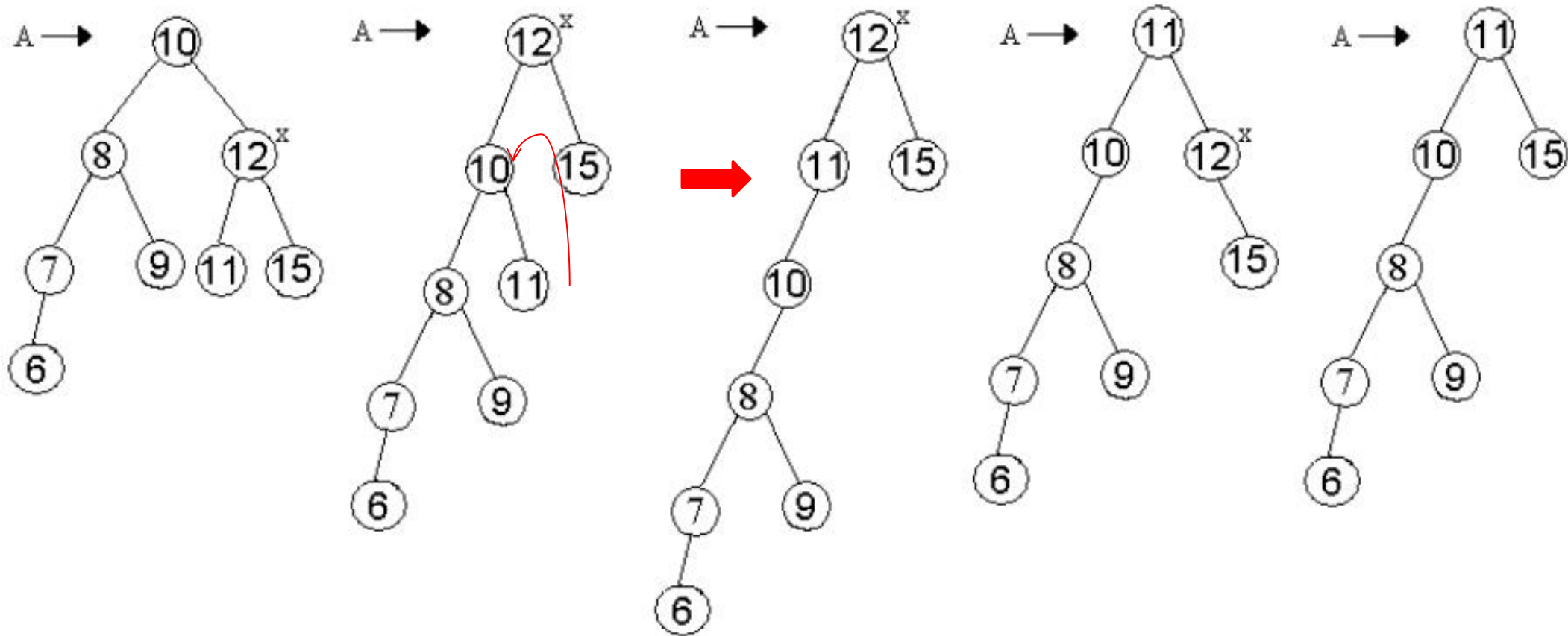
Senão faz splay do elemento menor mais próximo de x



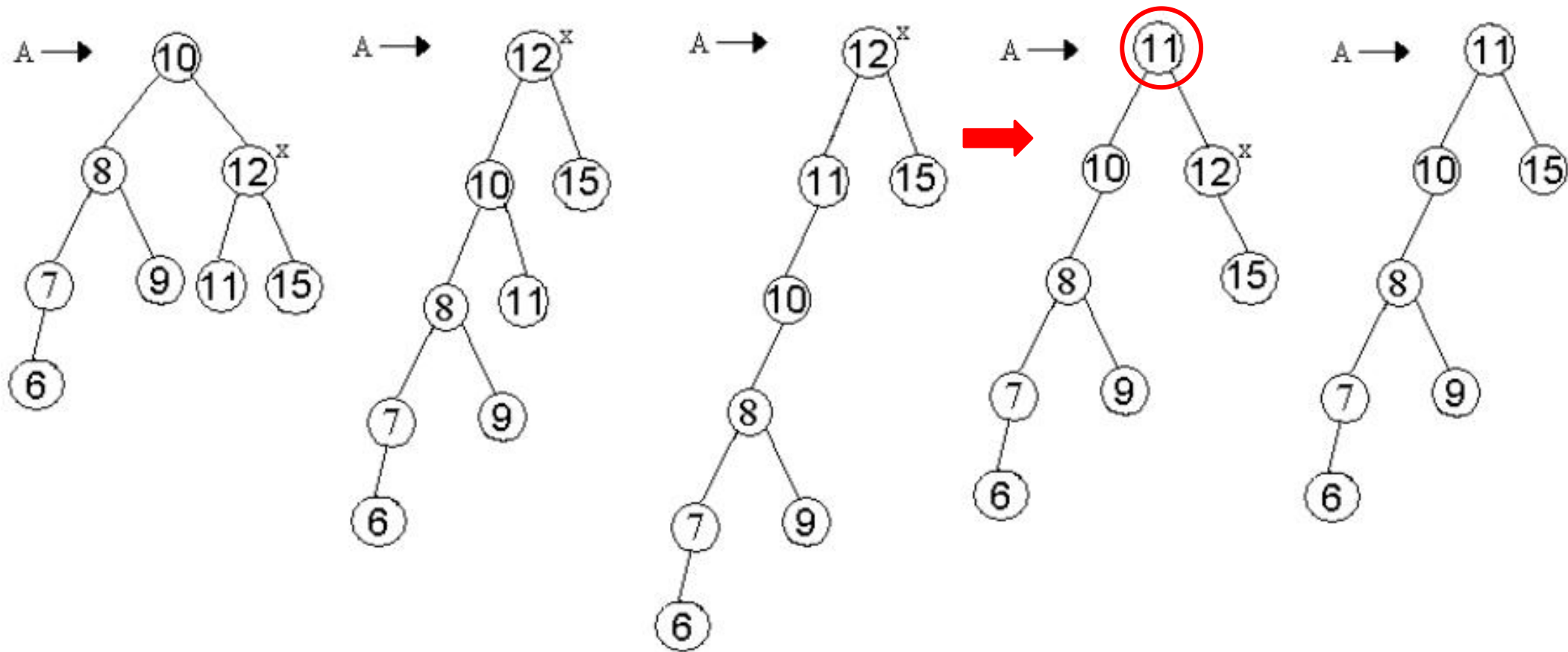
- Pesquisa o elemento 12



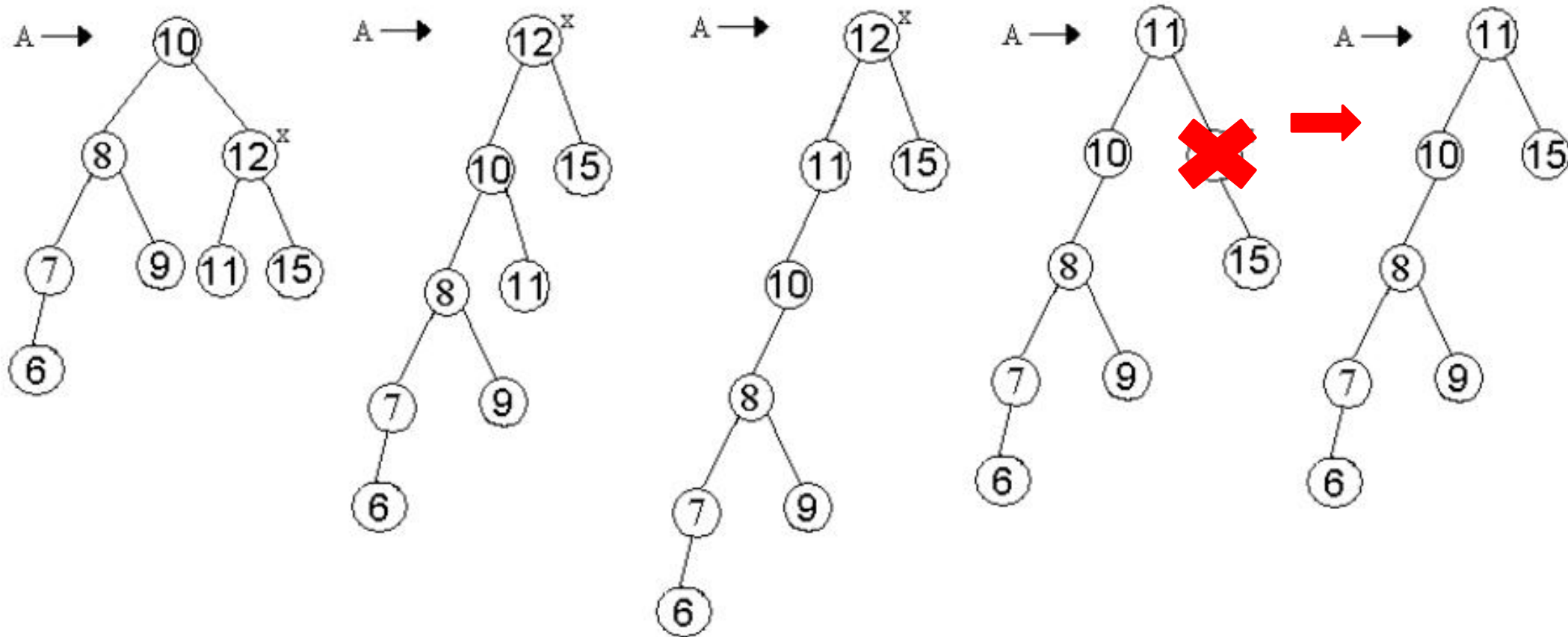
– SPLAY (12, A)



– SPLAY (12, A') onde A' é a sub-árvore esquerda do 12



- O elemento menor mais próximo de 12 vai para a raiz e sem filho esquerdo, podemos então remover



- Eliminar o 12 e ligar o pai de x (12) com seu filho direito (15)

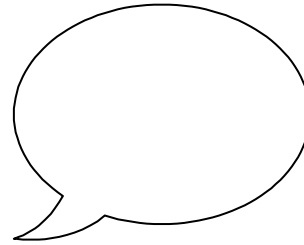
Árvores Balanceadas: Splay

- ❑ Árvores do tipo splay podem ficar desbalanceadas.
- ❑ Mas garantem uma complexidade $O(\log n)$ ao longo do tempo de utilização.
- ❑ Uma sequência qualquer de m operações demora, no pior caso $O(m \log n)$.

Árvores Balanceadas: Splay

□ Aplicações:

- Hospital: registro de pacientes recentes vai para raiz no momento da internação e permanece por alguma tempo.
- Sistema de arquivos: Microsoft Windows utiliza na sua indexação de arquivos por árvore splay.



Dúvidas?



franciny@ufj.edu.br

*Todos os avisos, aulas, anotações, exercícios, avaliações e notas são disponibilizados
no SIGAA*