

# Unidade VI:

## Balanceamento de Árvores Binárias

Prof. Max do Val Machado



**PUC Minas**

Instituto de Ciências Exatas e Informática  
Curso de Ciência da Computação

# Balanceamento de Árvores

- Qual é a vantagem de uma árvore estar balanceada?

# Balanceamento de Árvores

- Qual é a vantagem de uma árvore estar balanceada?
  - Resposta: eficiência em termos de pesquisa, inserção e remoção
- Inicialmente, toda árvore é balanceada e elas podem desbalancear após as operações de inserção e remoção

# Ideia Básica do Balanceamento de Árvores

- As árvores desbalanceadas para a esquerda devem ser rotacionadas para a direita e as para a direita, para a esquerda

# Tipos de Rotação

- Rotação simples à esquerda
- Rotação simples à direita
- Rotação dupla direita – esquerda
- Rotação dupla esquerda - direita

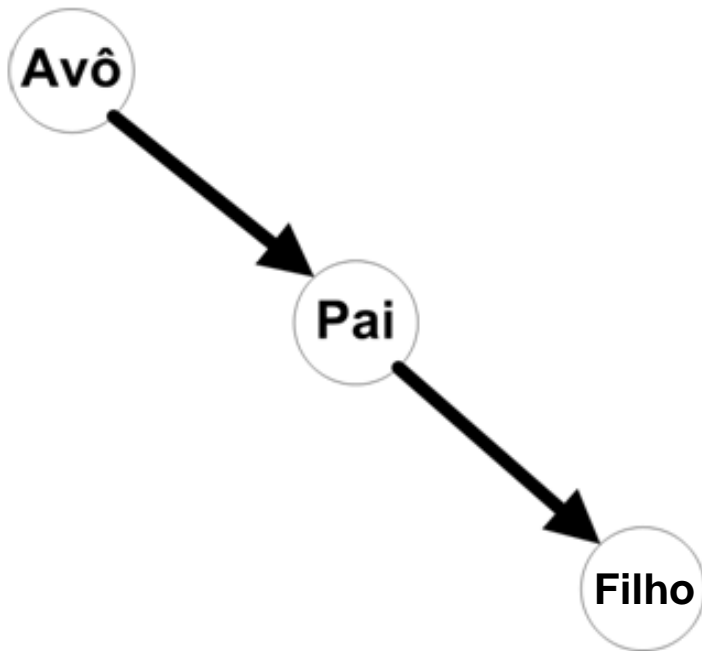
# Tipos de Rotação

- Rotação simples à esquerda
- Rotação simples à direita
- Rotação dupla direita – esquerda
- Rotação dupla esquerda - direita

Quando usar cada uma delas e como fazer ?

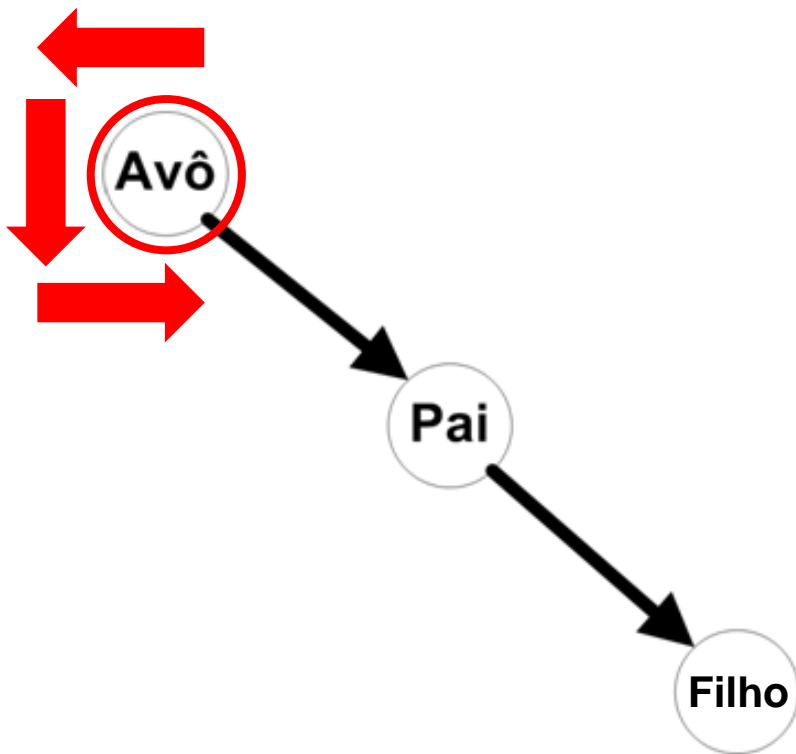
# Rotação Simples à Esquerda

- Usada em subárvores em que o pai e o filho estão desbalanceados para a direita



# Rotação Simples à Esquerda

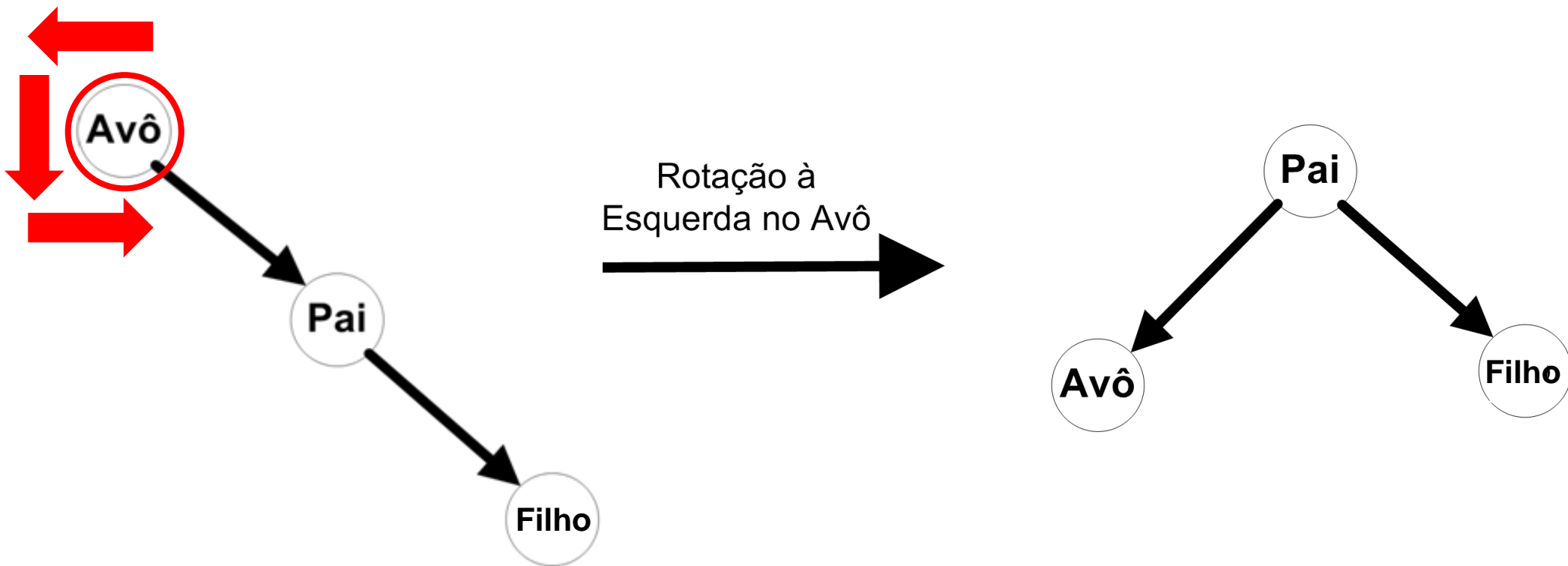
- Usada em subárvores em que o pai e o filho estão desbalanceados para a direita



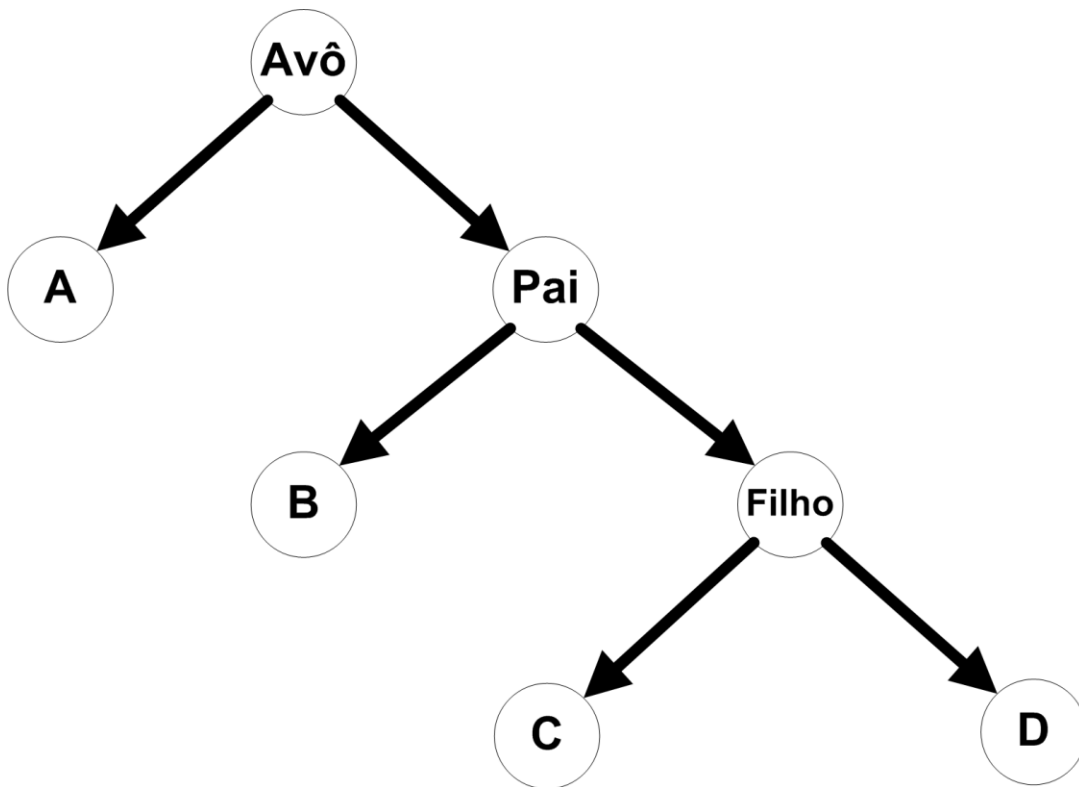


# Rotação Simples à Esquerda

- Usada em subárvores em que o pai e o filho estão desbalanceados para a direita

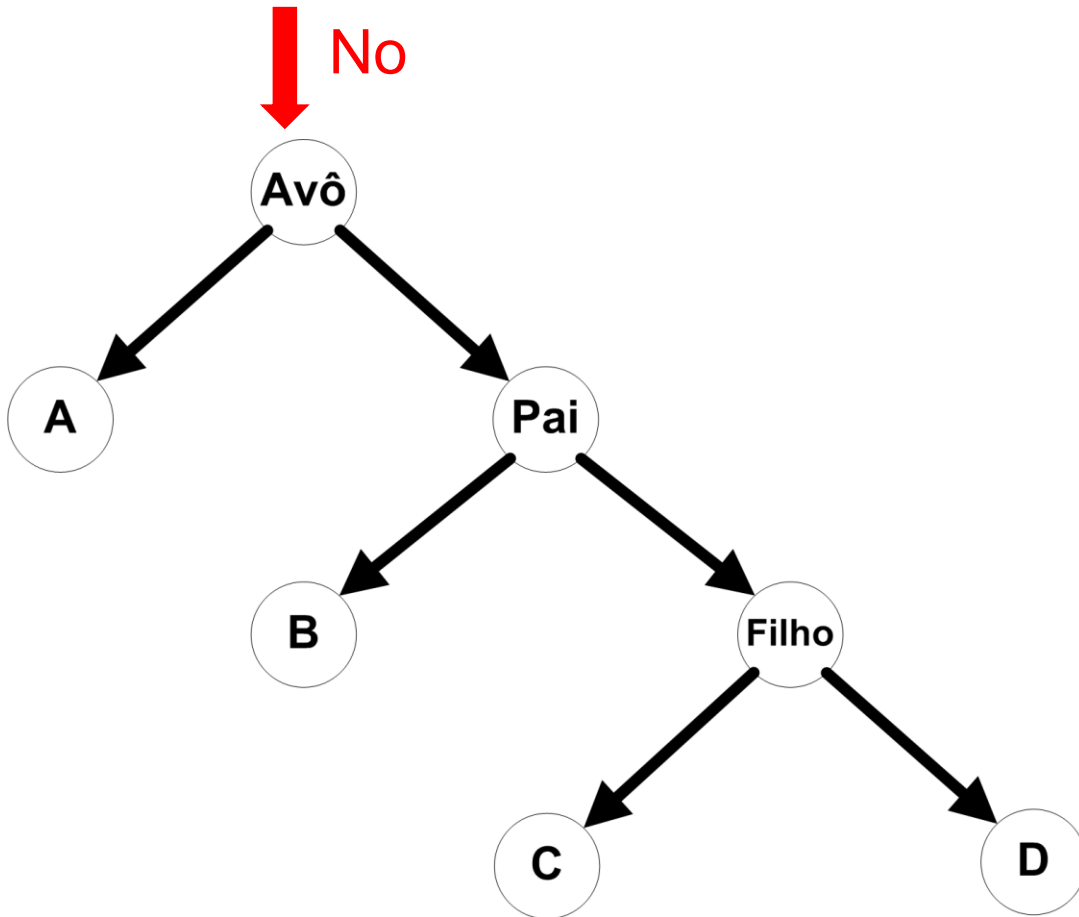


# Implementação da Rotação à Esquerda



```
No rotacionarEsq (No no) {  
  No noDir = no.dir;  
  No noDirEsq = noDir.esq;  
  
  noDir.esq = no;  
  no.dir = noDirEsq;  
  
  return noDir;  
}
```

# Implementação da Rotação à Esquerda



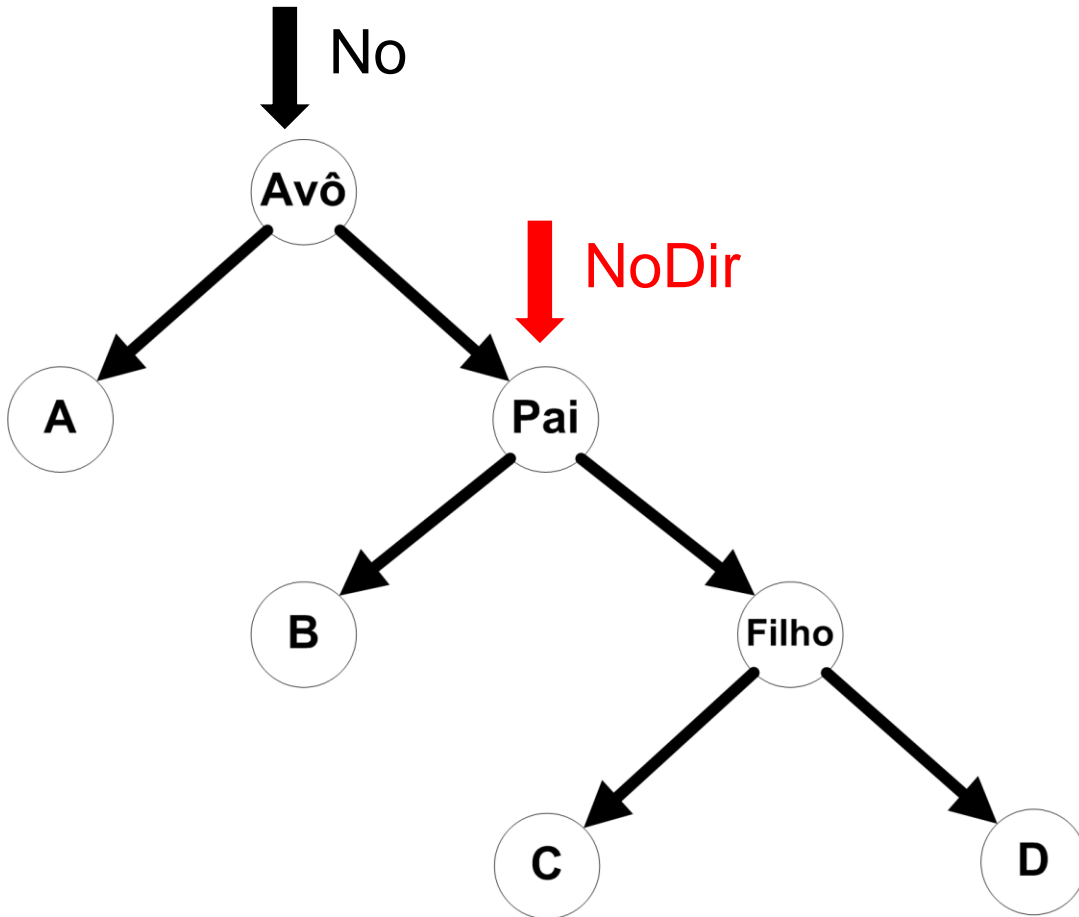
```

No rotacionarEsq (No no) {
    No noDir = no.dir;
    No noDirEsq = noDir.esq;

    noDir.esq = no;
    no.dir = noDirEsq;

    return noDir;
}
    
```

# Implementação da Rotação à Esquerda



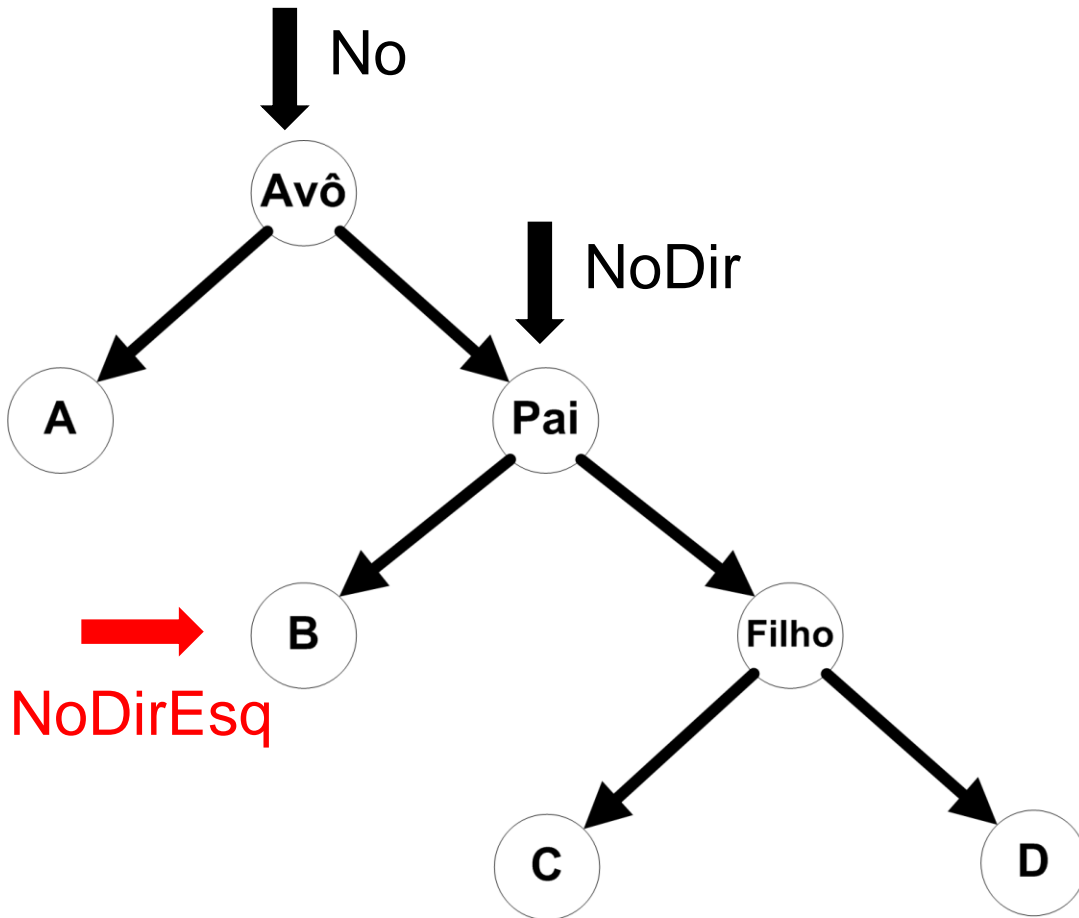
```

No rotacionarEsq (No no) {
    No noDir = no.dir;
    No noDirEsq = noDir.esq;

    noDir.esq = no;
    no.dir = noDirEsq;

    return noDir;
}
    
```

# Implementação da Rotação à Esquerda



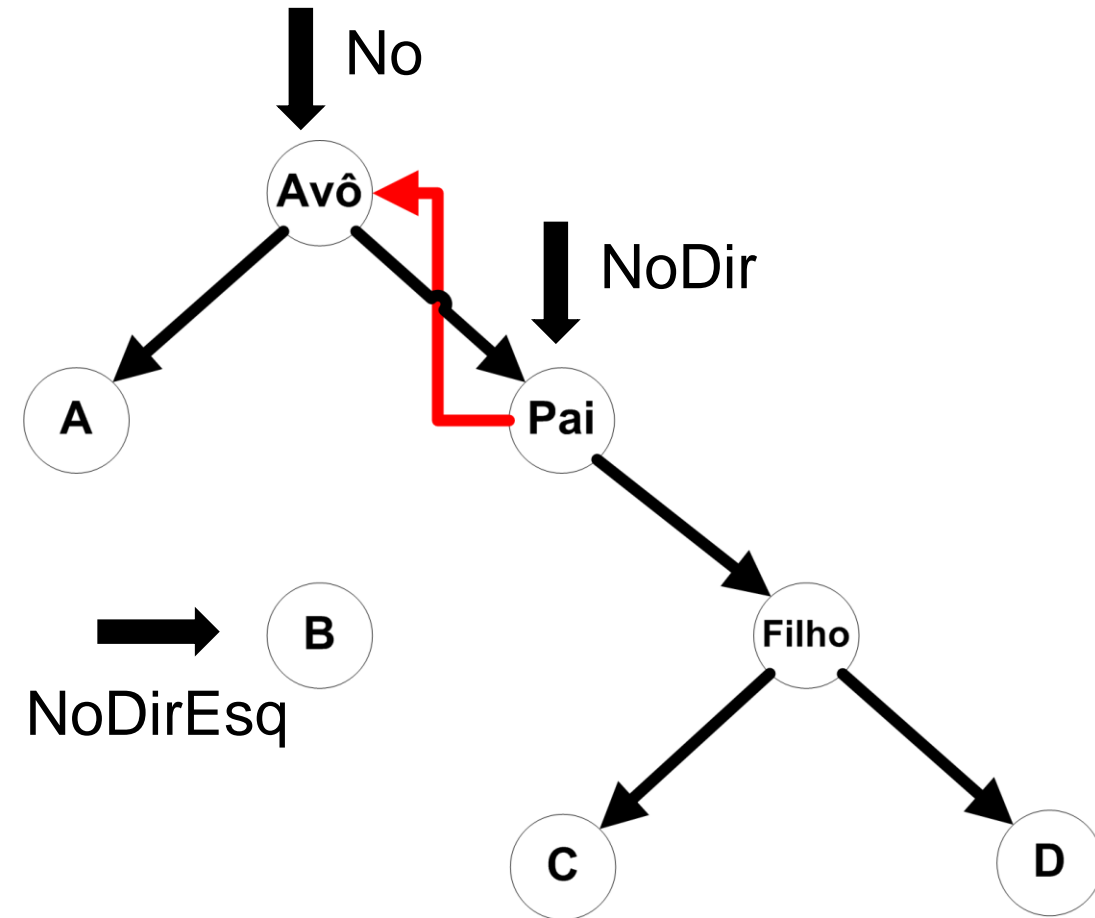
```

No rotacionarEsq (No no) {
  No noDir = no.dir;
  No noDirEsq = noDir.esq;

  noDir.esq = no;
  no.dir = noDirEsq;

  return noDir;
}
    
```

# Implementação da Rotação à Esquerda



```

No rotacionarEsq (No no) {
  No noDir = no.dir;
  No noDirEsq = noDir.esq;

```

```

  noDir.esq = no;
  no.dir = noDirEsq;

```

```

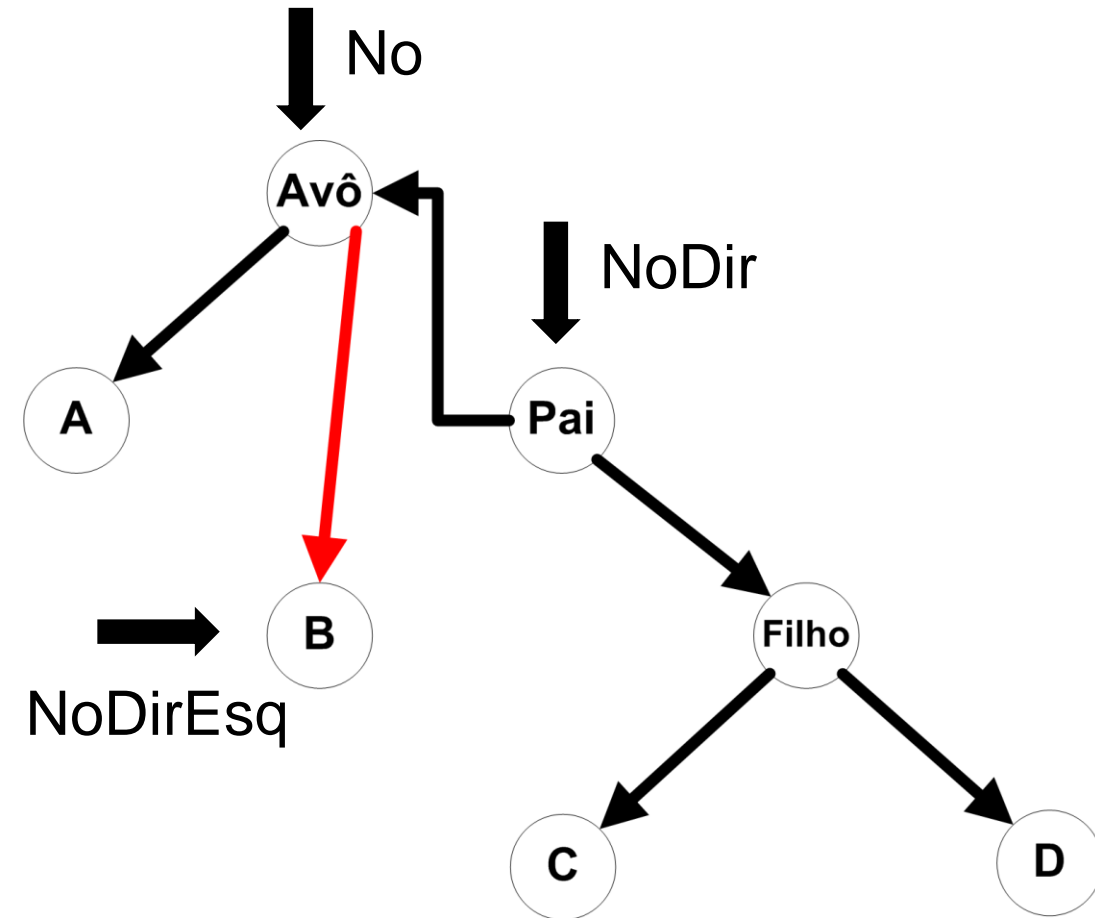
  return noDir;

```

```

}
```

# Implementação da Rotação à Esquerda

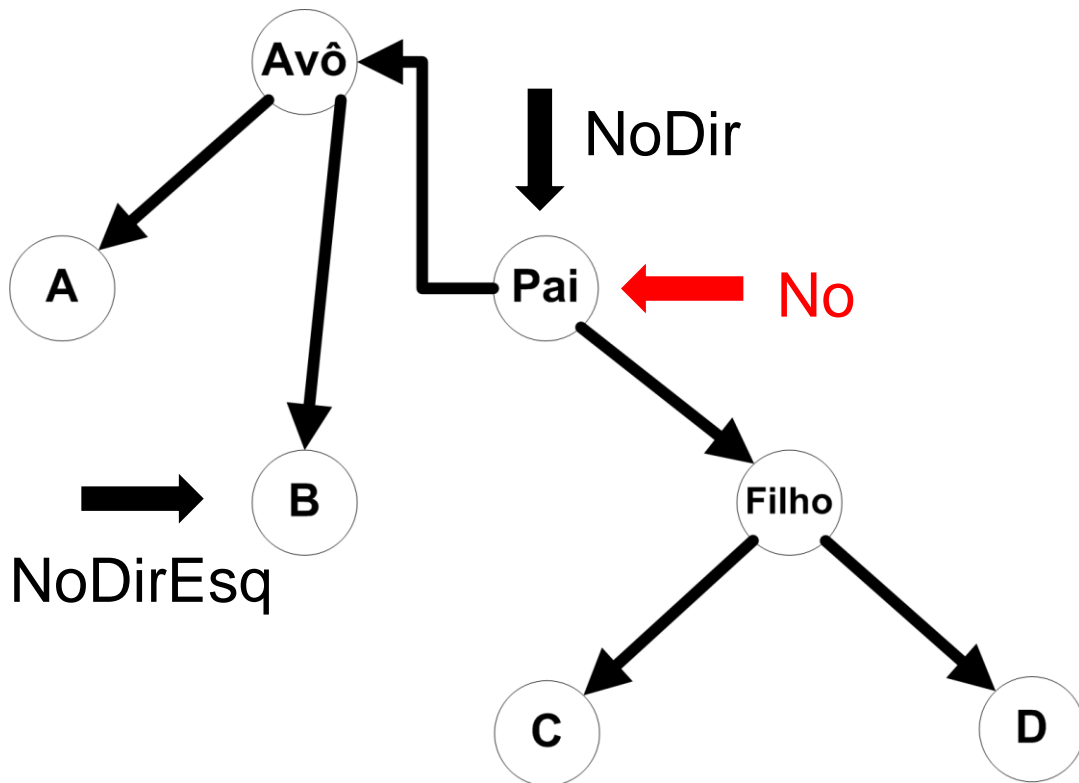


```
No rotacionarEsq (No no) {
  No noDir = no.dir;
  No noDirEsq = noDir.esq;

  noDir.esq = no;
  no.dir = noDirEsq;

  return noDir;
}
```

# Implementação da Rotação à Esquerda



```

No rotacionarEsq (No no) {
  No noDir = no.dir;
  No noDirEsq = noDir.esq;

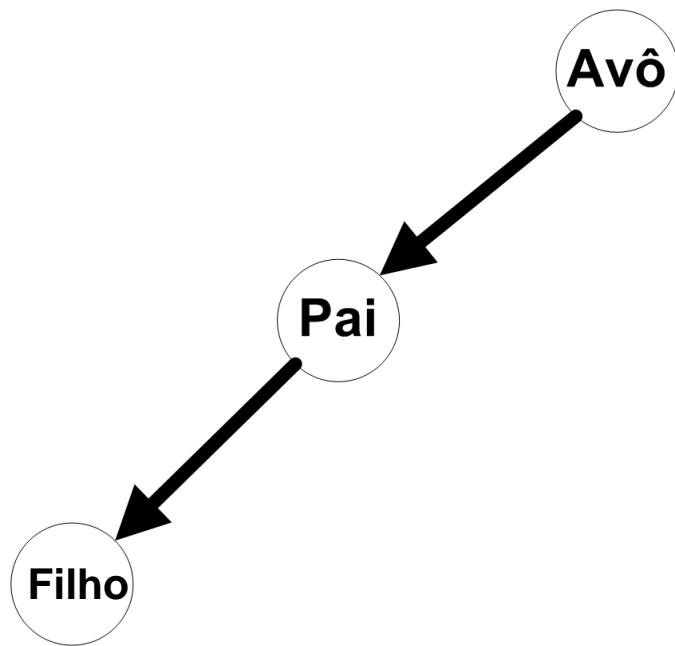
  noDir.esq = no;
  no.dir = noDirEsq;

  return noDir;
}
    
```



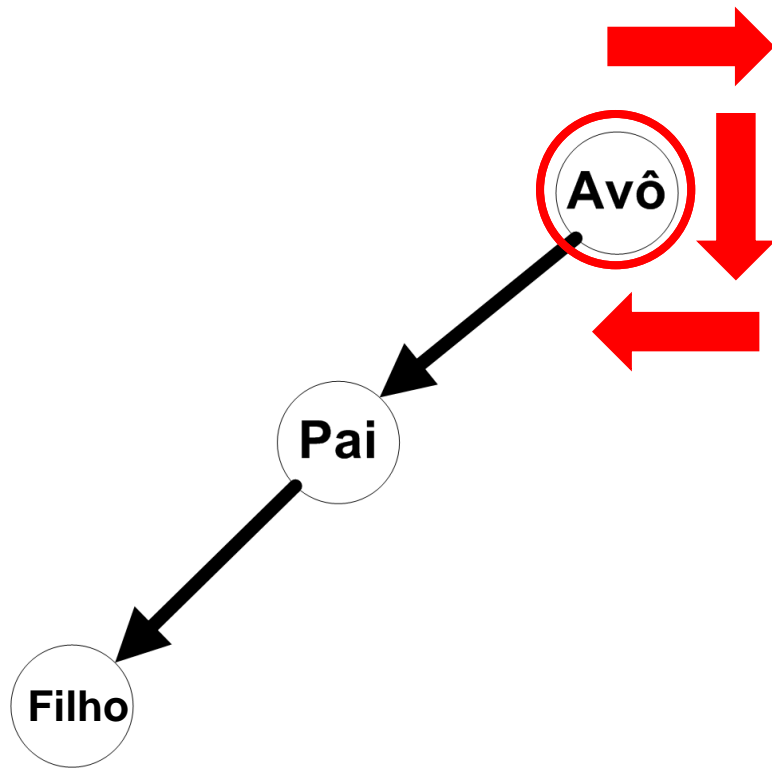
# Rotação Simples à Direita

- Usada em subárvores em que o pai e o filho estão desbalanceados para a esquerda



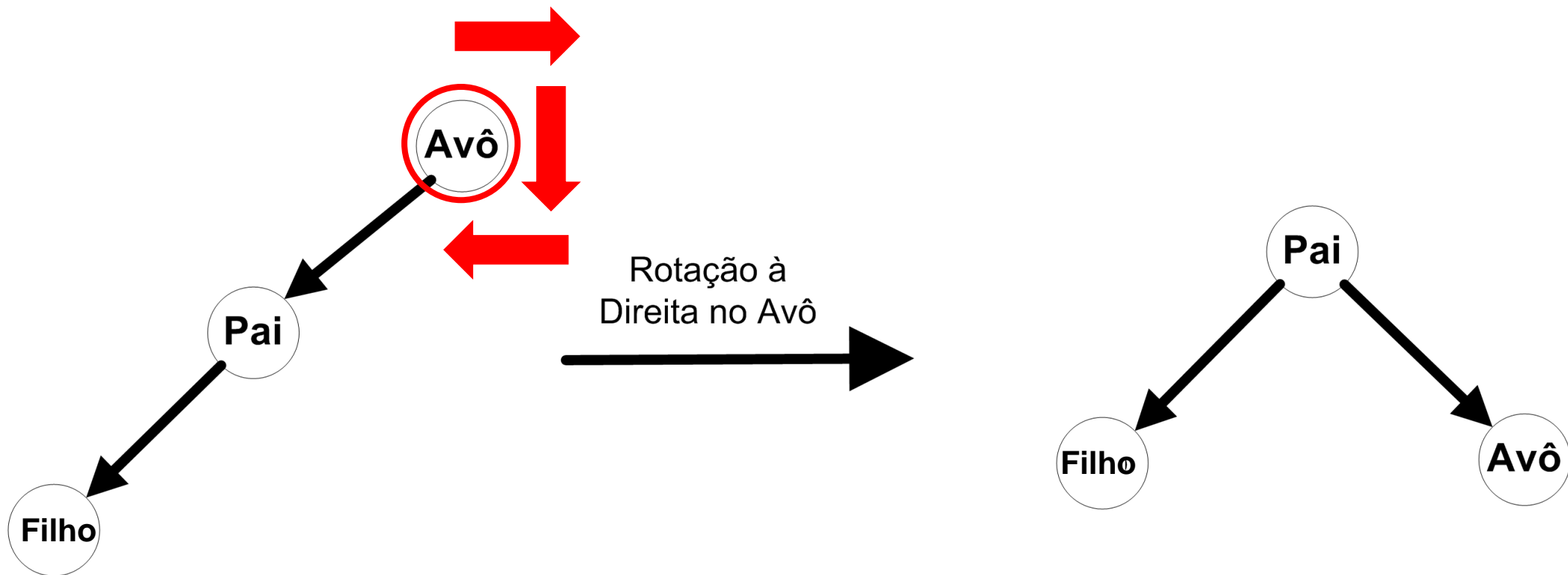
# Rotação Simples à Direita

- Usada em subárvores em que o pai e o filho estão desbalanceados para a esquerda

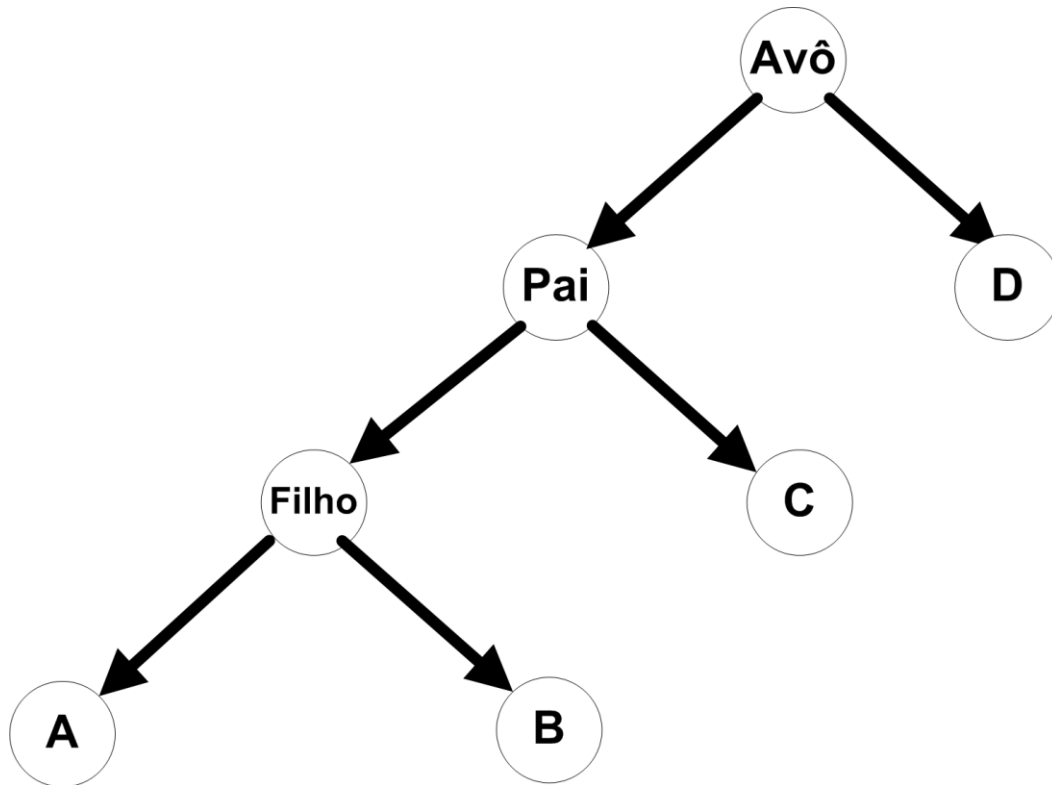


# Rotação Simples à Direita

- Usada em subárvores em que o pai e o filho estão desbalanceados para a esquerda

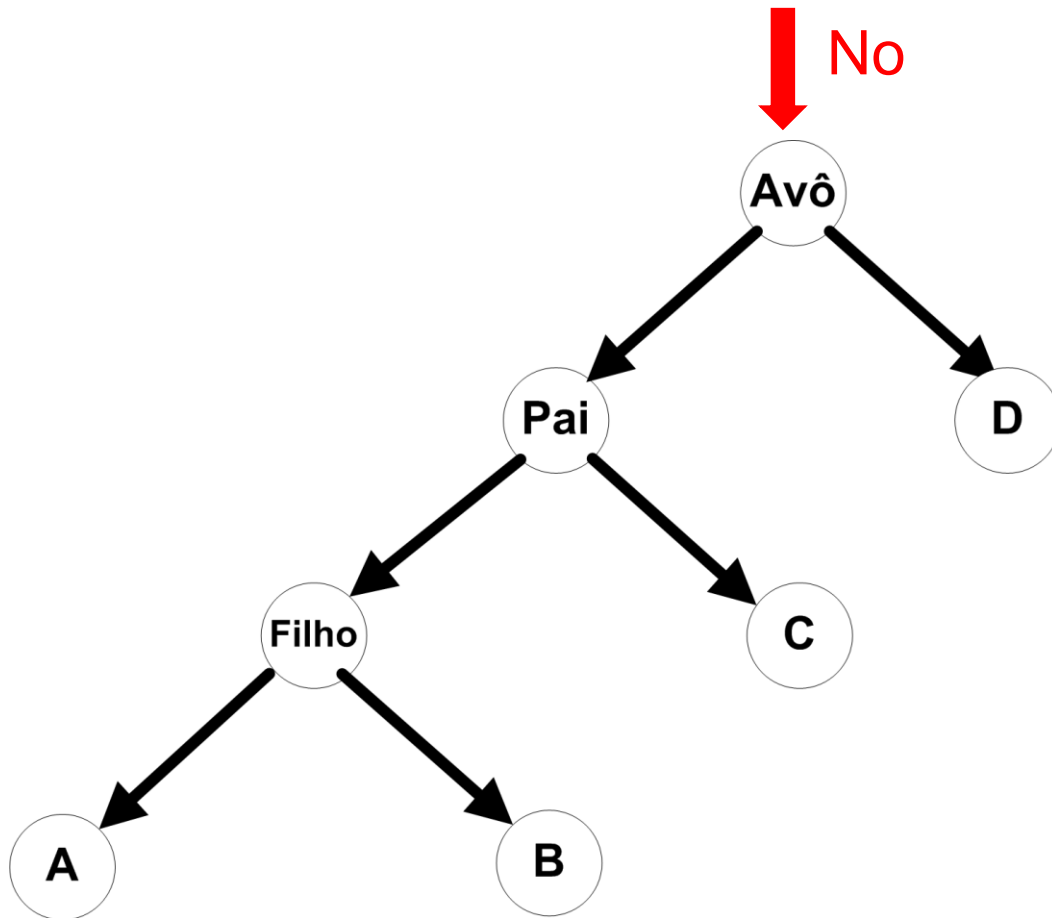


# Implementação da Rotação à Direita



```
No rotacionarDir (No no) {  
  No noEsq = no.esq;  
  No noEsqDir = noEsq.dir;  
  
  noEsq.dir = no;  
  no.esq = noEsqDir;  
  
  return noEsq;  
}
```

# Implementação da Rotação à Direita



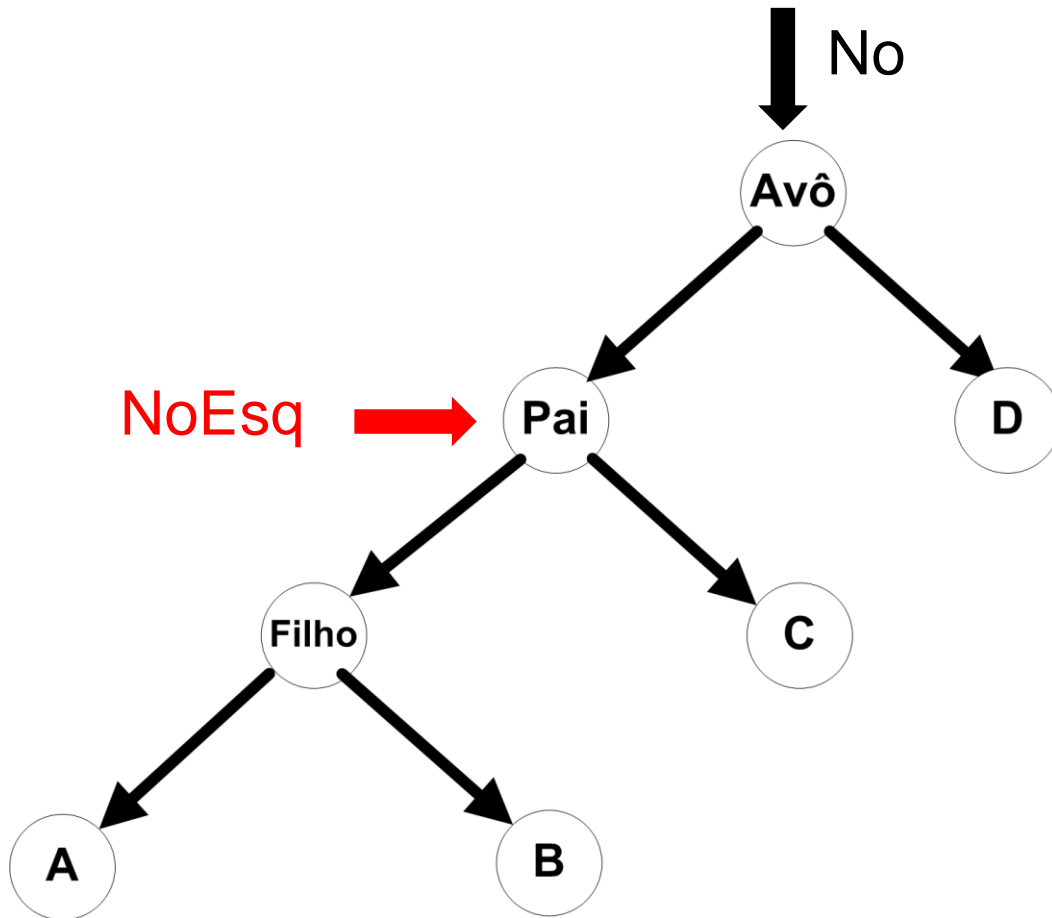
```

No rotacionarDir (No no) {
    No noEsq = no.esq;
    No noEsqDir = noEsq.dir;

    noEsq.dir = no;
    no.esq = noEsqDir;

    return noEsq;
}
    
```

# Implementação da Rotação à Direita



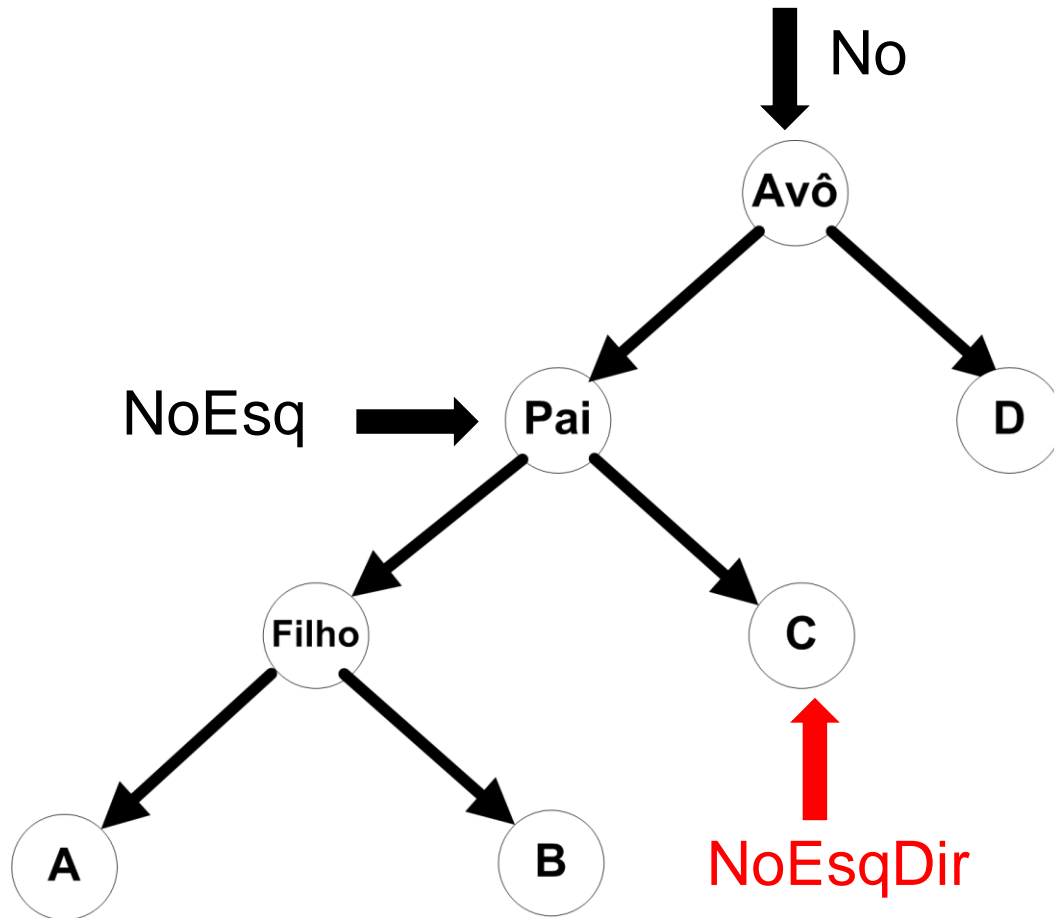
```

No rotacionarDir (No no) {
    No noEsq = no.esq;
    No noEsqDir = noEsq.dir;

    noEsq.dir = no;
    no.esq = noEsqDir;

    return noEsq;
}
    
```

# Implementação da Rotação à Direita



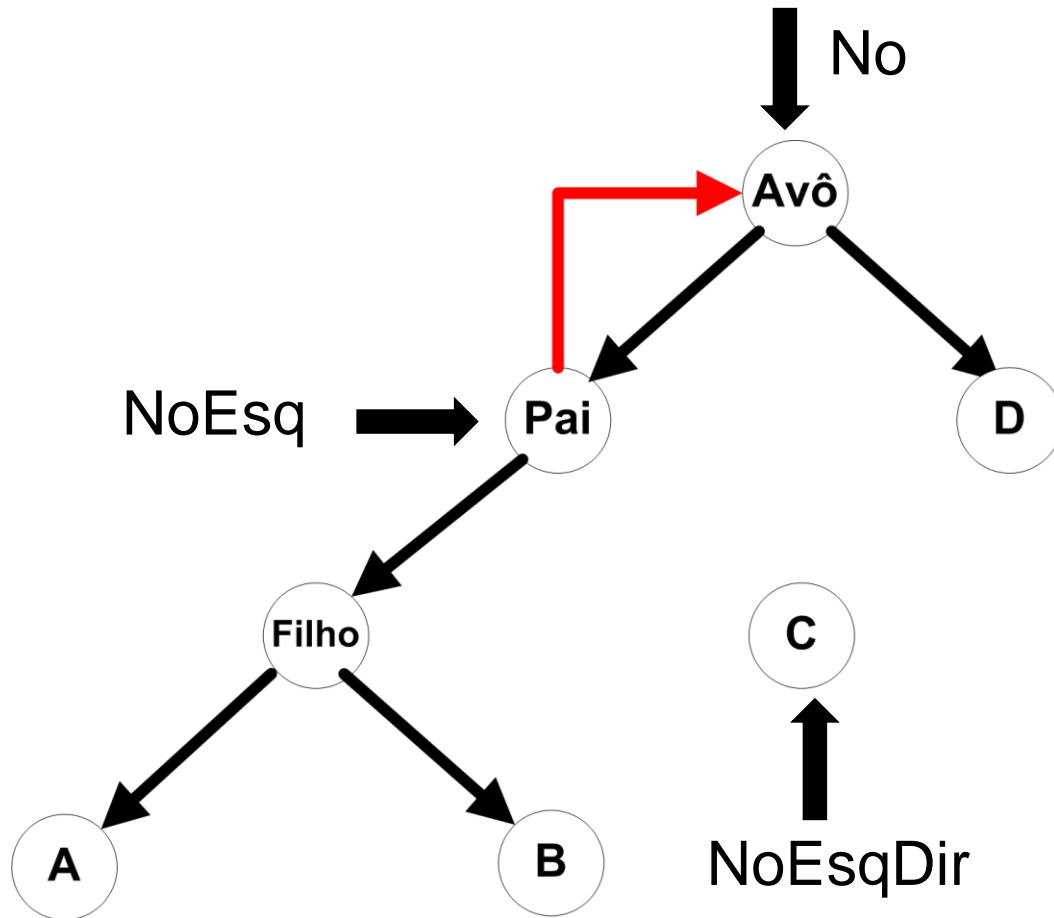
```

No rotacionarDir (No no) {
  No noEsq = no.esq;
  No noEsqDir = noEsq.dir;

  noEsq.dir = no;
  no.esq = noEsqDir;

  return noEsq;
}
    
```

# Implementação da Rotação à Direita



```

No rotacionarDir (No no) {
  No noEsq = no.esq;
  No noEsqDir = noEsq.dir;

```

```

  noEsq.dir = no;
  no.esq = noEsqDir;

```

```

  return noEsq;

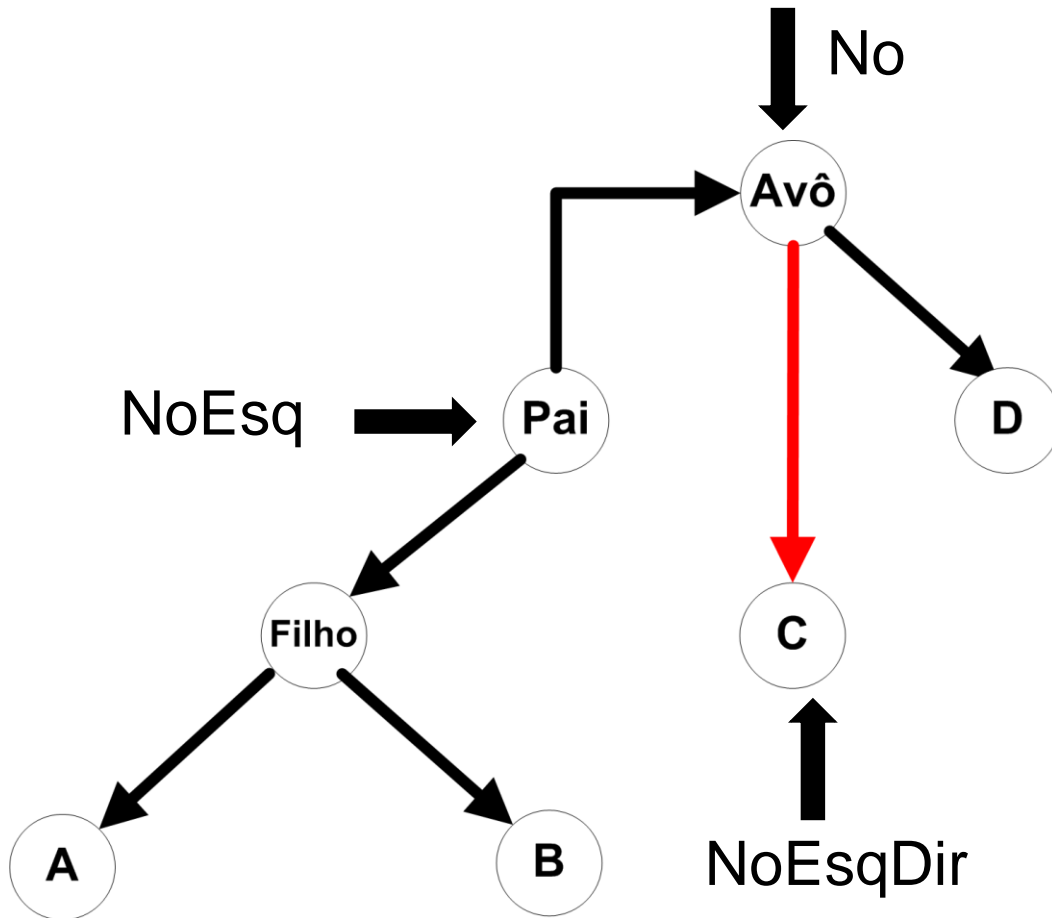
```

```

}
```



# Implementação da Rotação à Direita



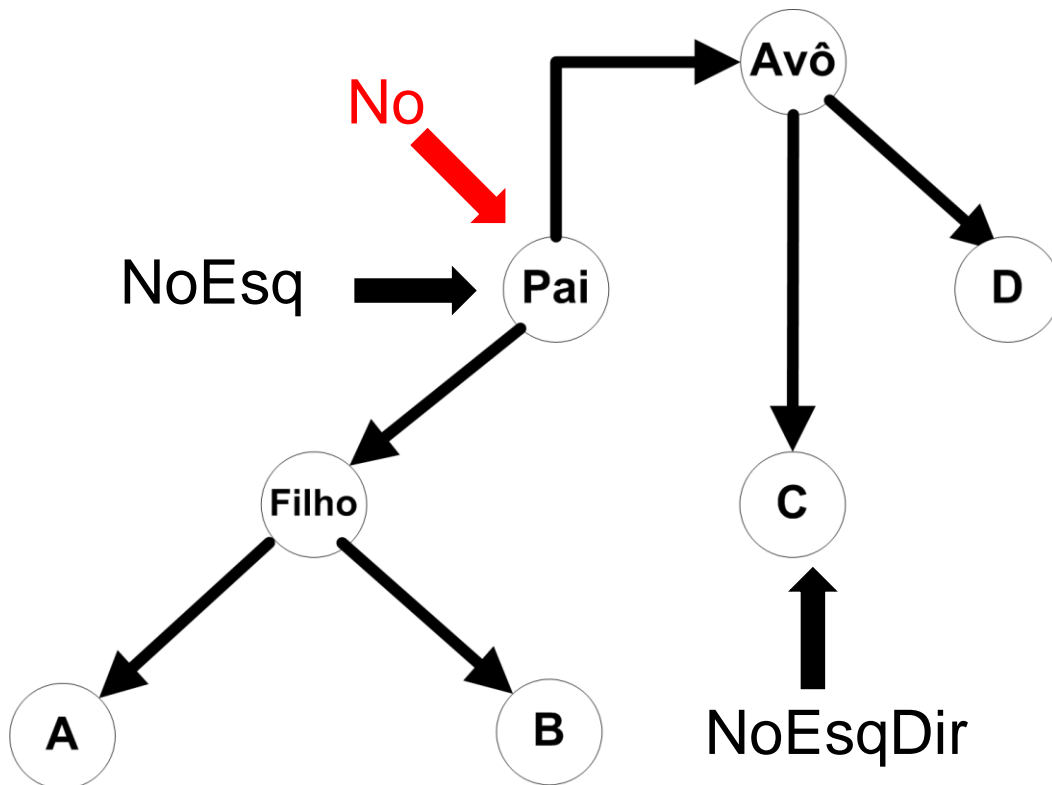
```

No rotacionarDir (No no) {
  No noEsq = no.esq;
  No noEsqDir = noEsq.dir;

  noEsq.dir = no;
  no.esq = noEsqDir;

  return noEsq;
}
    
```

# Implementação da Rotação à Direita



```

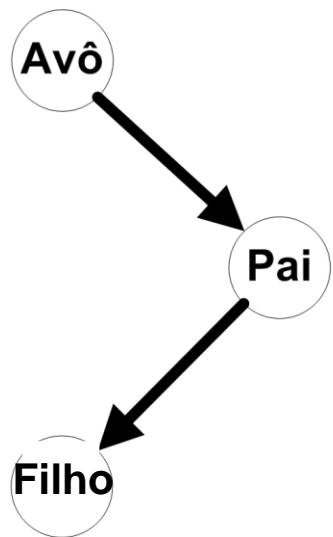
No rotacionarDir (No no) {
  No noEsq = no.esq;
  No noEsqDir = noEsq.dir;

  noEsq.dir = no;
  no.esq = noEsqDir;

  return noEsq;
}
    
```

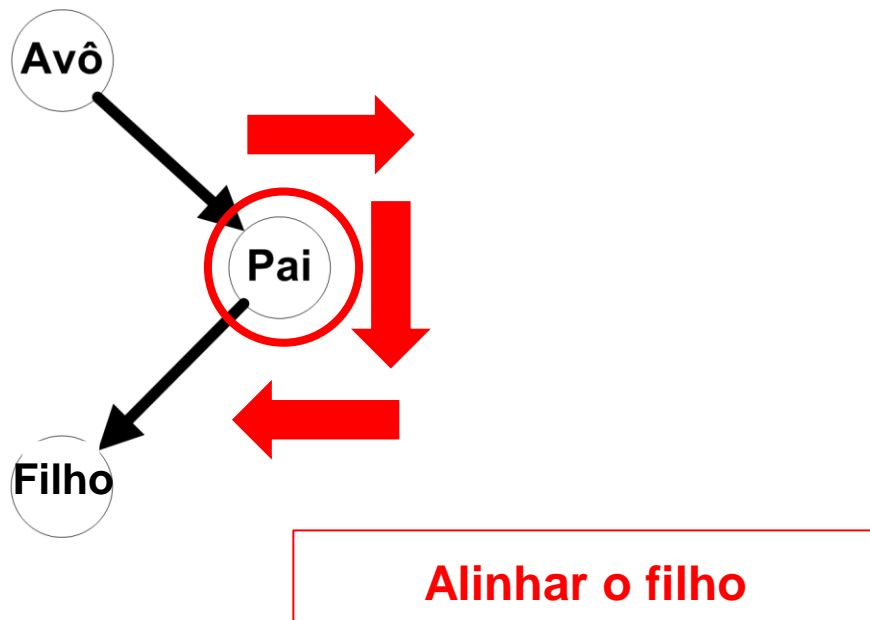
# Rotação Dupla Direita – Esquerda

- Usada em subárvores em que o pai está desbalanceado para a direita e o filho para a esquerda



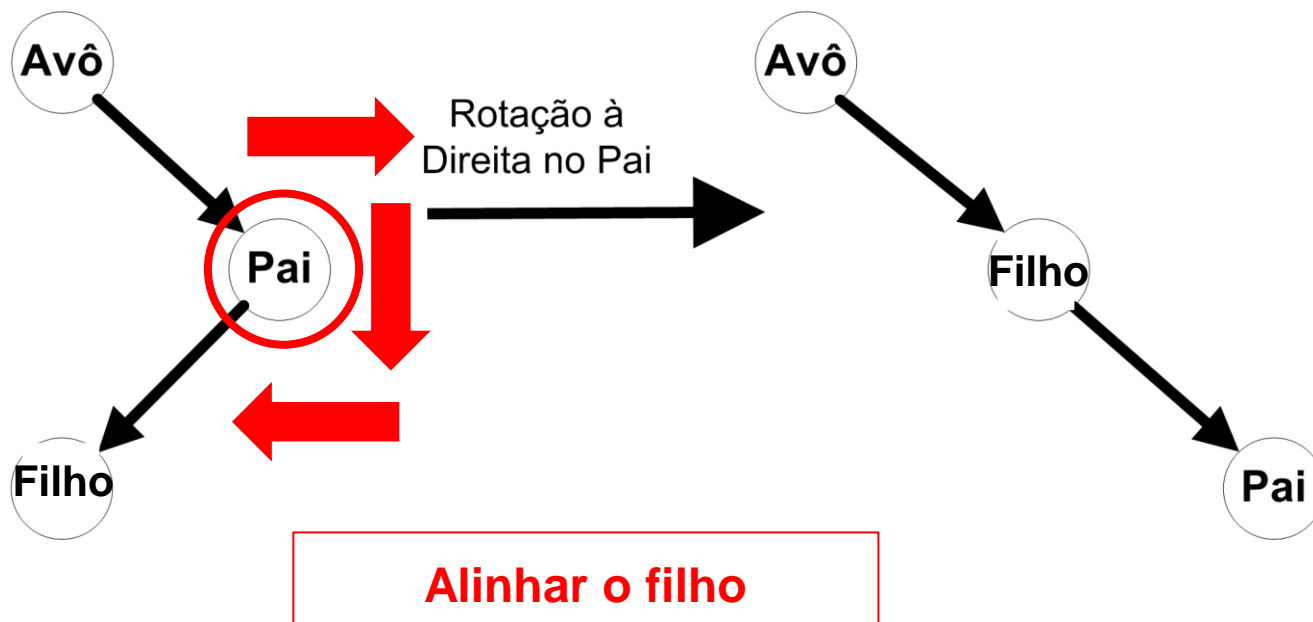
# Rotação Dupla Direita – Esquerda

- Usada em subárvores em que o pai está desbalanceado para a direita e o filho para a esquerda



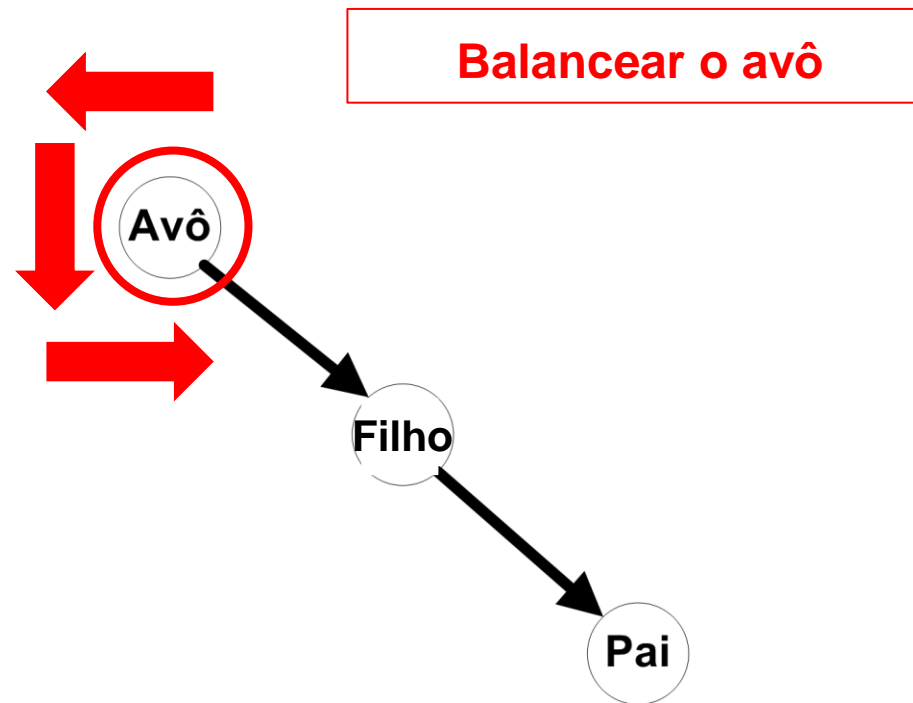
# Rotação Dupla Direita – Esquerda

- Usada em subárvores em que o pai está desbalanceado para a direita e o filho para a esquerda



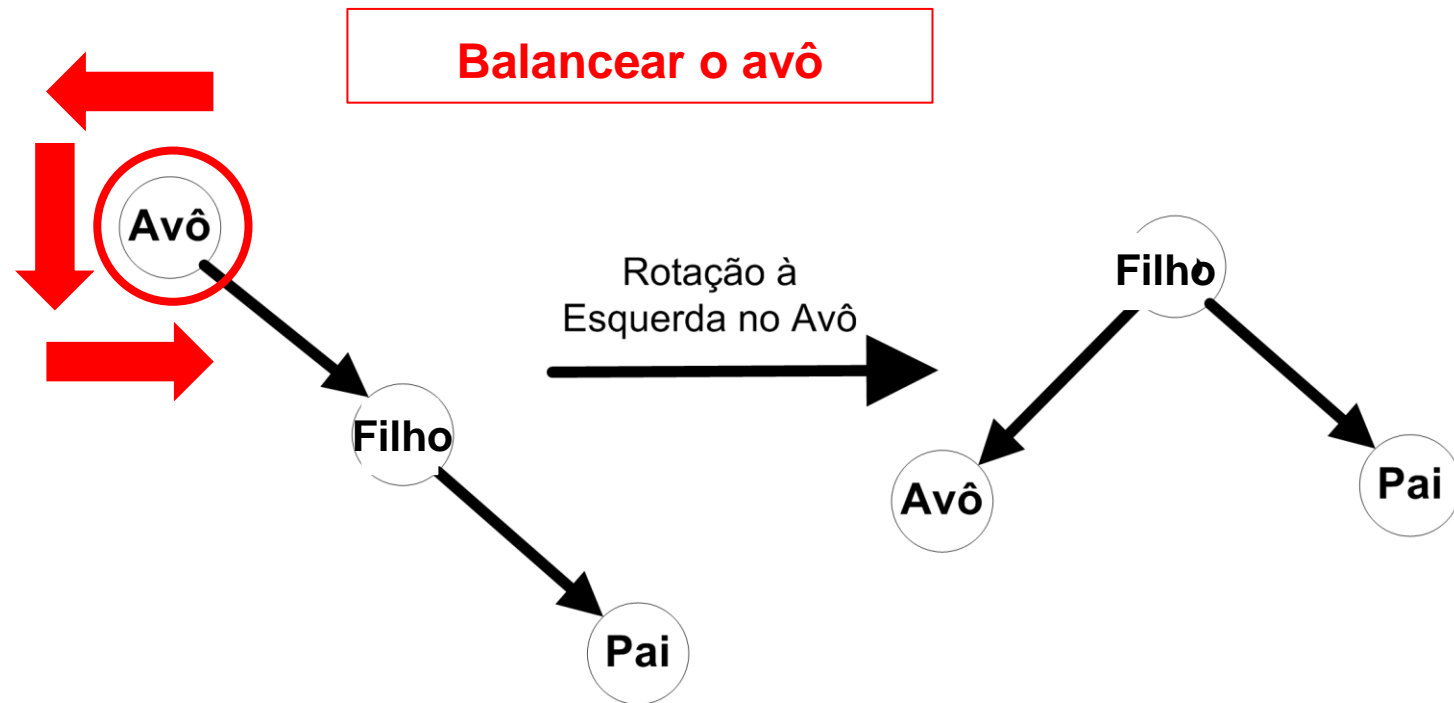
# Rotação Dupla Direita – Esquerda

- Usada em subárvores em que o pai está desbalanceado para a direita e o filho para a esquerda



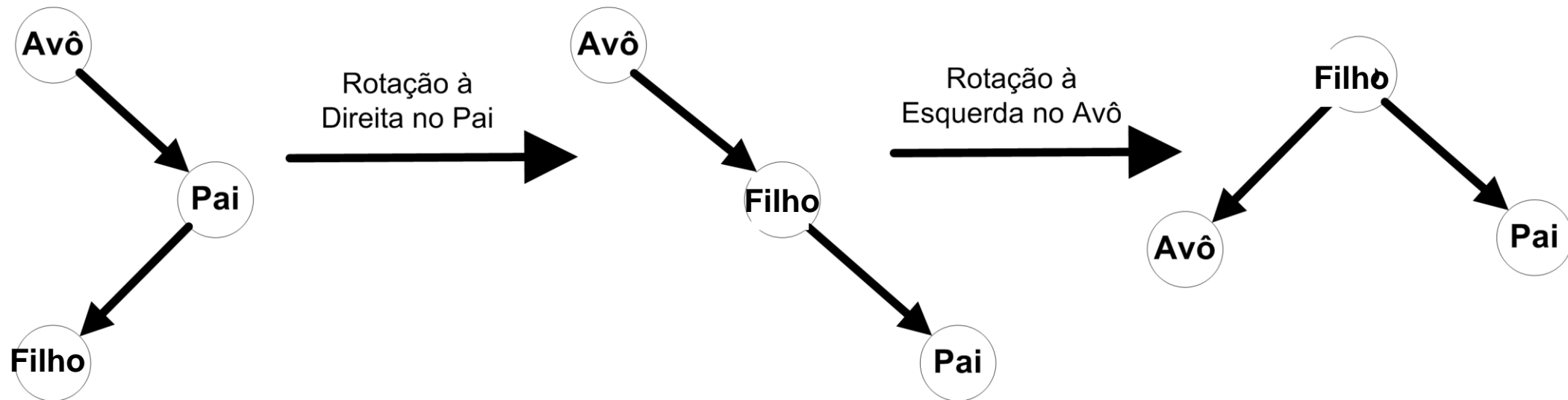
# Rotação Dupla Direita – Esquerda

- Usada em subárvores em que o pai está desbalanceado para a direita e o filho para a esquerda



# Implementação da Rotação à Direita - Esquerda

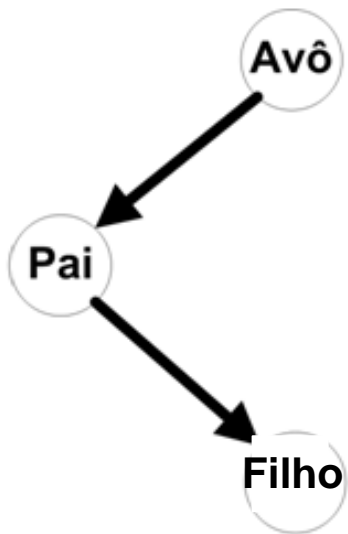
```
No rotacionarDirEsq (No no) {  
    no.dir = rotacionarDir (no.dir);  
    return rotacionarEsq(no);  
}
```





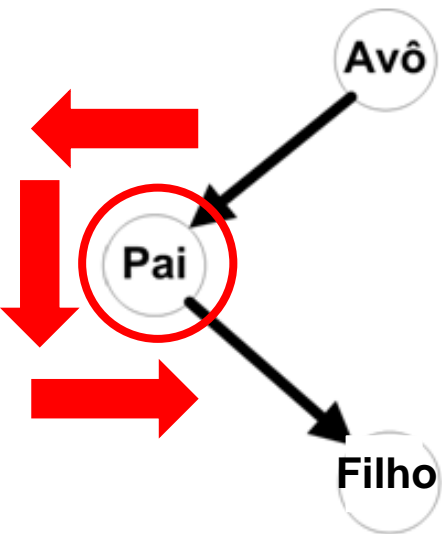
# Rotação Dupla Esquerda – Direita

- Usada em subárvores em que o pai está desbalanceado para a esquerda e o filho para a direita



# Rotação Dupla Esquerda – Direita

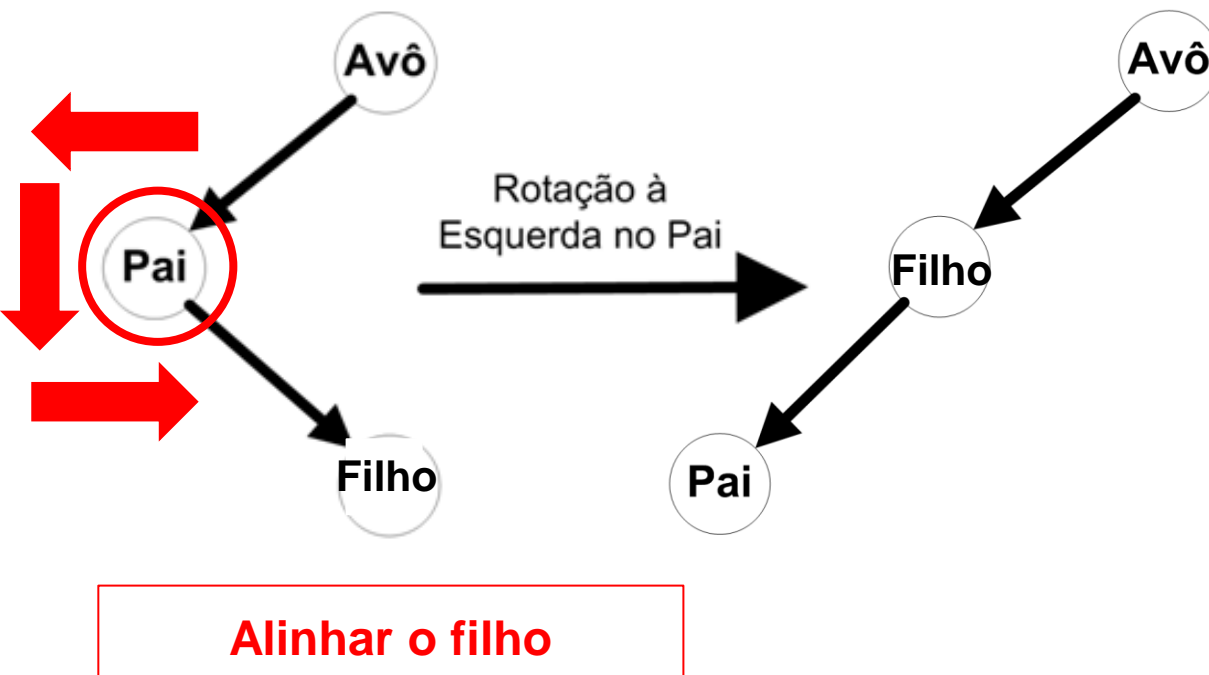
- Usada em subárvores em que o pai está desbalanceado para a esquerda e o filho para a direita



**Alinhar o filho**

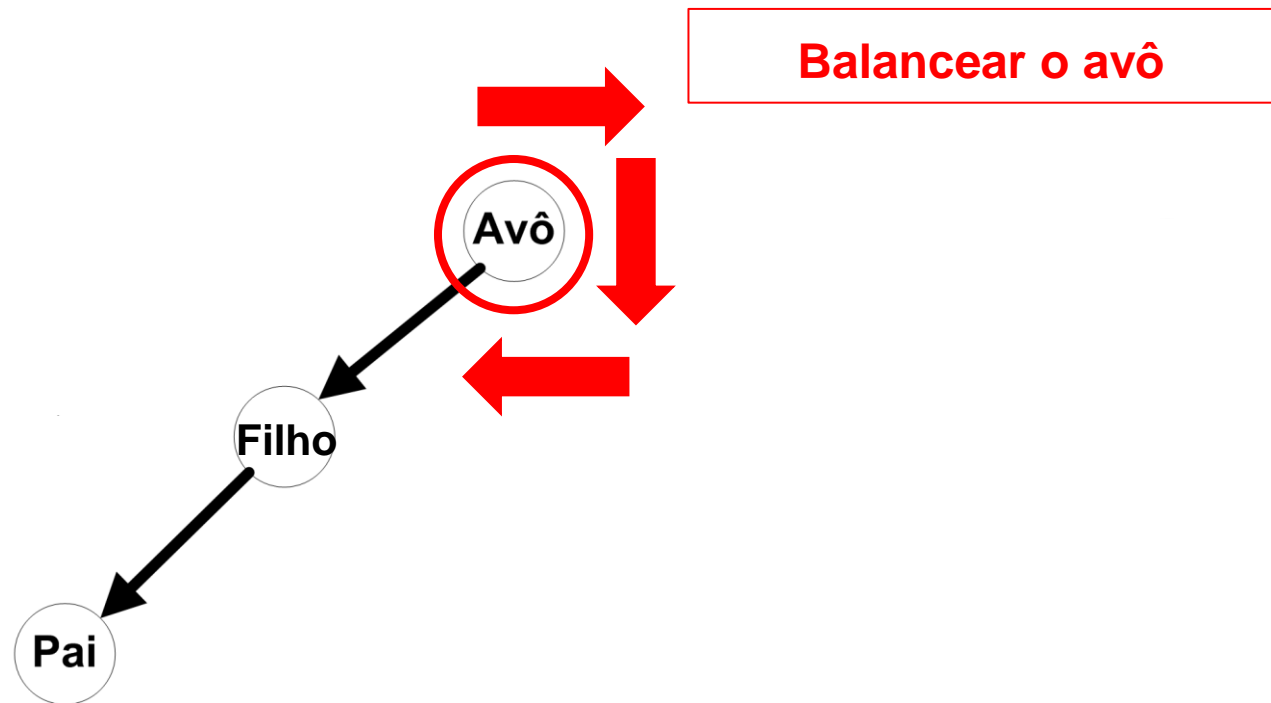
# Rotação Dupla Esquerda – Direita

- Usada em subárvores em que o pai está desbalanceado para a esquerda e o filho para a direita



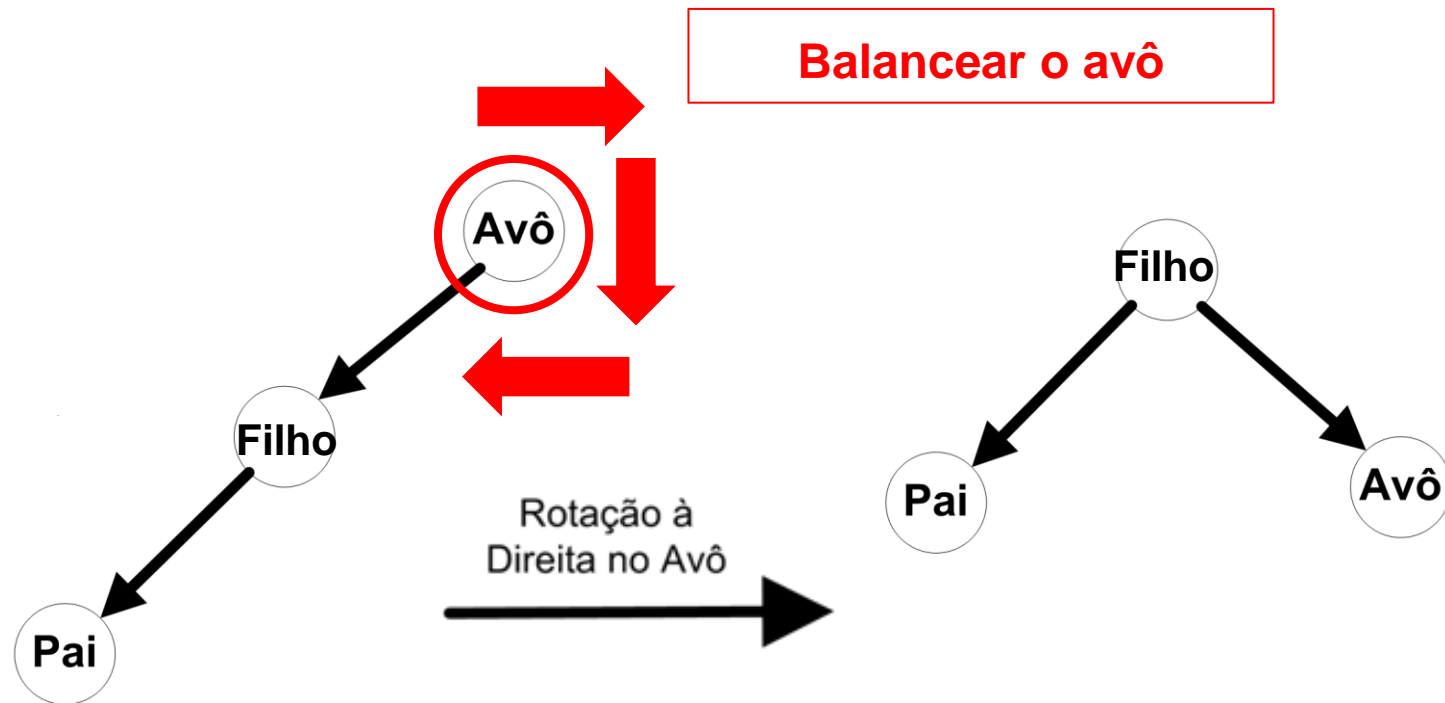
# Rotação Dupla Esquerda – Direita

- Usada em subárvores em que o pai está desbalanceado para a esquerda e o filho para a direita



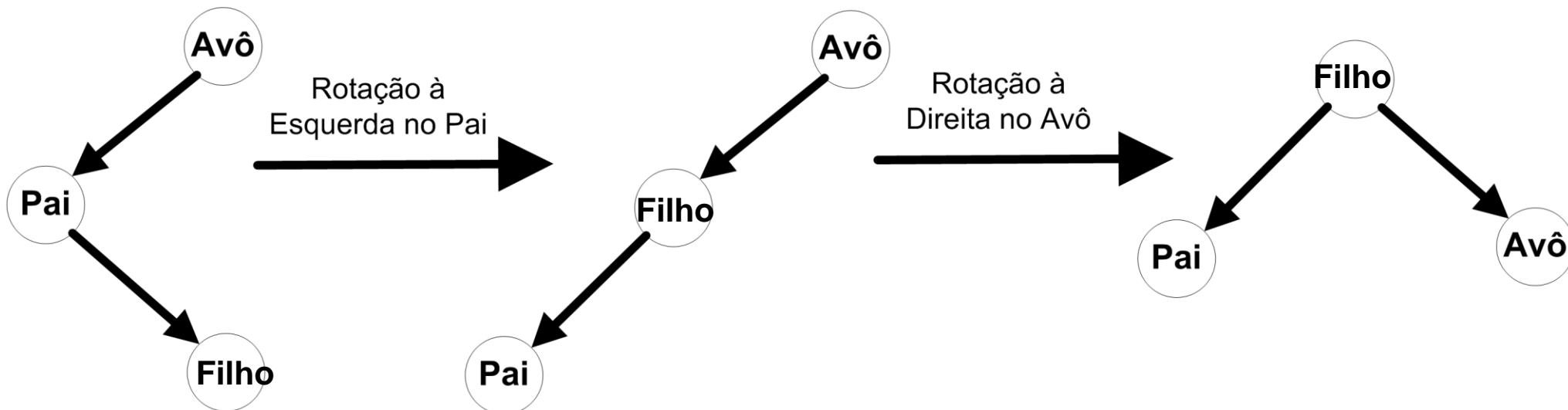
# Rotação Dupla Esquerda – Direita

- Usada em subárvores em que o pai está desbalanceado para a esquerda e o filho para a direita



# Implementação da Rotação à Esquerda - Direita

```
No rotacionarEsqDir (No no) {  
    no.esq = rotacionarEsq (no.esq);  
    return rotacionarDir(no);  
}
```

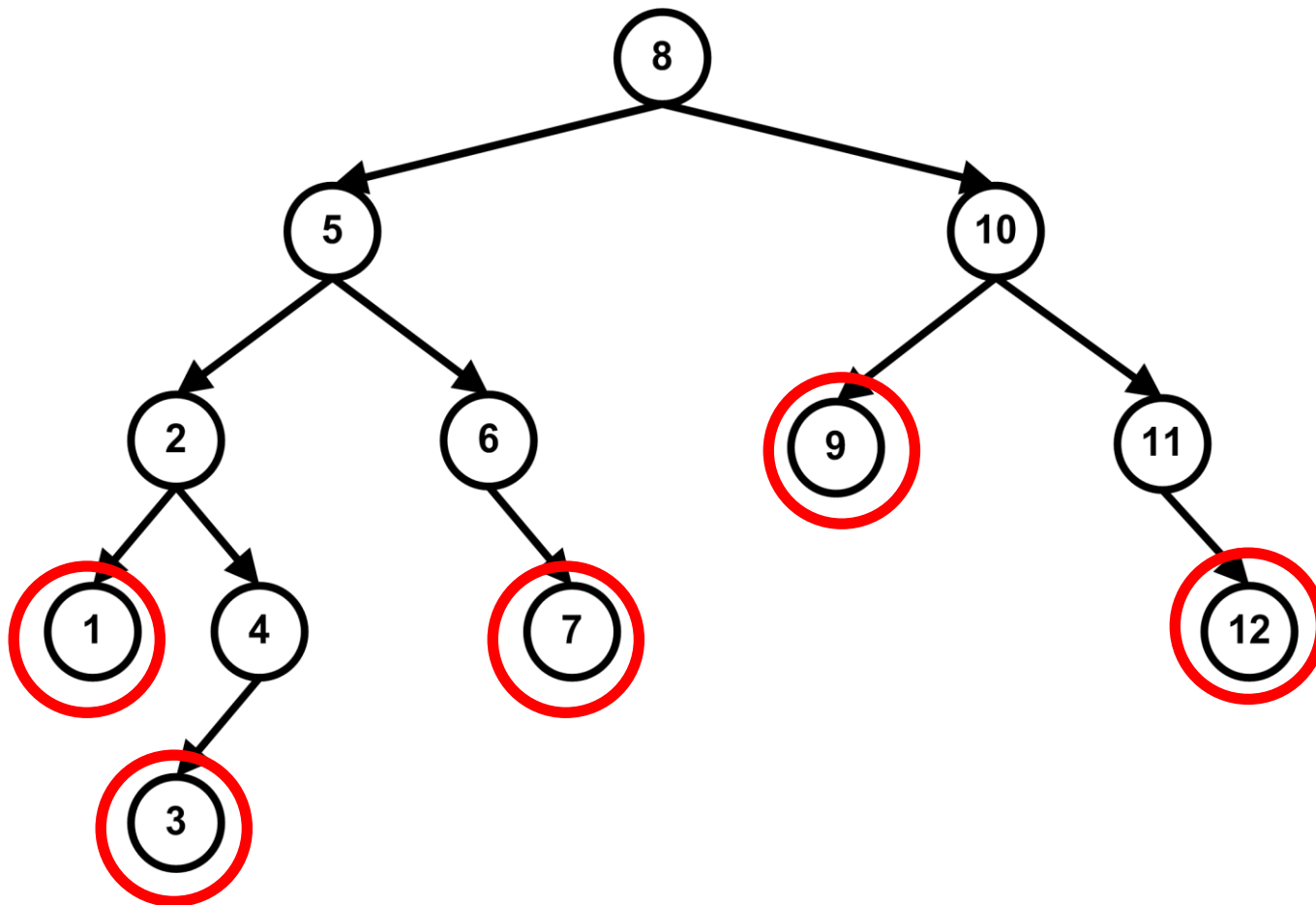


# Balanceamento de Árvores

- Qual é o custo para se manter uma árvore balanceada?
- Na prática, não existe “muita” diferença entre árvores balanceadas ou praticamente balanceadas
- Algumas árvores balanceadas como a AVL e a Alvinegra permitem árvores praticamente balanceadas

# Balanceamento de Árvores

- Exemplo de árvore AVL em que as folhas ocupam mais de dois níveis





# Balanceamento de Árvores

- Exemplo de árvore Alvinegra em que as folhas ocupam mais de dois níveis

