

# Aula anterior

- Balanceamento de Árvores Binárias
- Árvore AVL

# Árvore 2-3-4


---

Prof. Diego Silva Caldeira Rocha

---

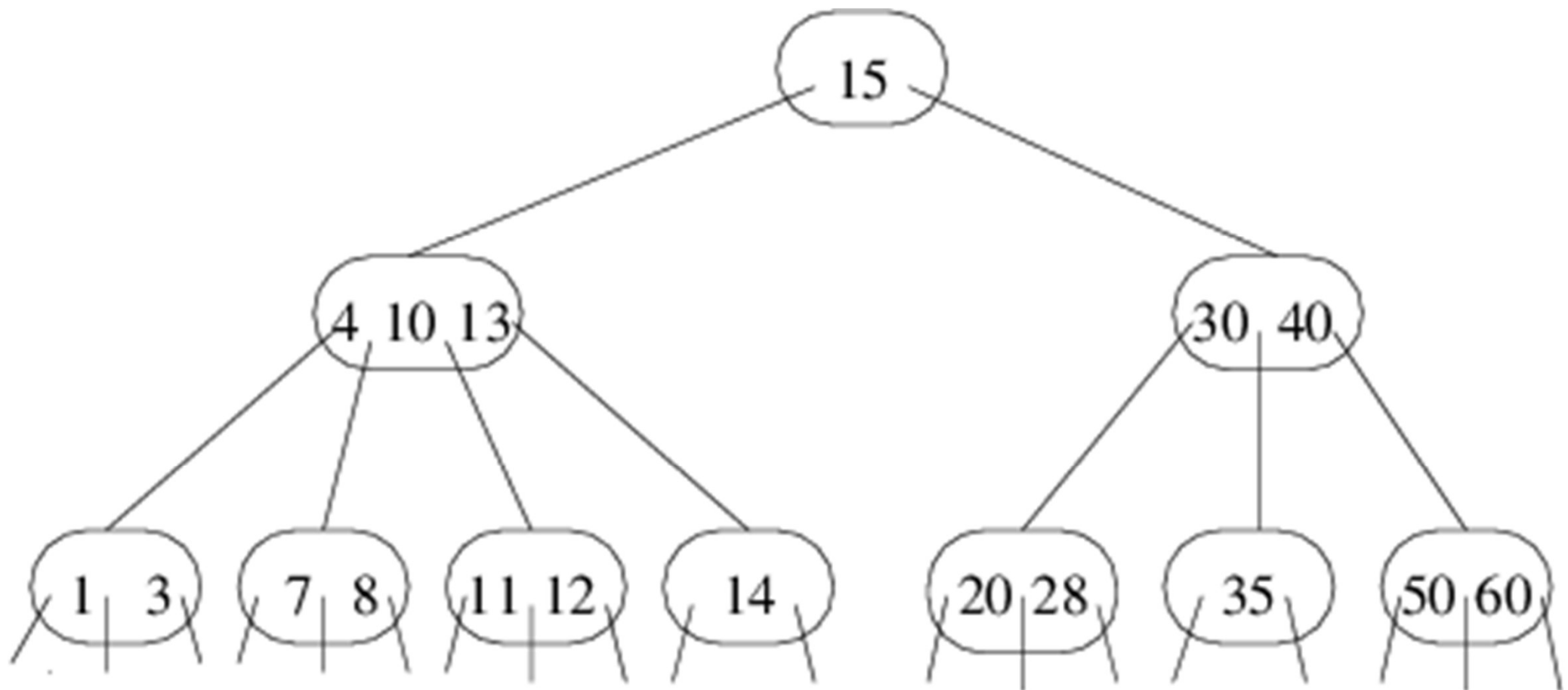
# Objetivos

- Introdução e Pesquisa
- Inserção com fragmentação por ascensão
- Inserção com fragmentação na descida

- **Introdução e Pesquisa** 
- Inserção com fragmentação por ascensão
- Inserção com fragmentação na descida

# Introdução

- Árvore de pesquisa cujos nós são de três tipos (2-nó, 3-nó ou 4-nó) e as folhas estão situadas no mesmo nível



# Propriedade das Árvore 2.3.4

- A altura  $h(n)$  de uma árvore-2.3.4 <sup>1</sup> contendo  $n$  itens é  $\Theta(\lg(n))$
- Por questões de simplificação, este material considera que todos os elementos da árvore são distintos

# Pesquisa em Árvores 2.3.4

- Similar ao das árvores binárias
- Funcionamento básico:
  - (1) Verificar se o elemento procurado  $x$  está no nó raiz
  - (2) Se estiver, tem-se uma **resposta positiva**
  - (3) Senão, se  $x < x_1$ , verificar na subárvore da esquerda
  - (4) Senão, se  $x < x_2$ , na do meio à esquerda
  - (5) Senão, se  $x < x_3$ , na do meio à direita
  - (6) Senão, na da direita
  - (7) Se a subárvore for nula, tem-se uma **resposta negativa**

# Análise de Complexidade da Pesquisa

- Número de comparações em uma pesquisa com sucesso:
  - Melhor Caso:  $\Theta(1)$
  - Pior e Caso Médio:  $\Theta(\lg(n))$



# Inserção

- Este material considera a inserção sempre nas folhas
- Se a folha for 2-nó ou 3-nó (1 ou 2 elementos), insere-se nela
- Senão (4-nó), fragmentamos a folha e reorganizando seus elementos. As duas técnicas para realizar essas tarefas são:
  - Inserção com fragmentação por ascensão
  - Inserção com fragmentação na descida

- Introdução e Pesquisa
- **Inserção com fragmentação por ascensão** ←
- Inserção com fragmentação na descida

# Inserção com Fragmentação na Ascensão

- Apresentaremos esta técnica fazendo, como exemplo, a inserção dos números 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

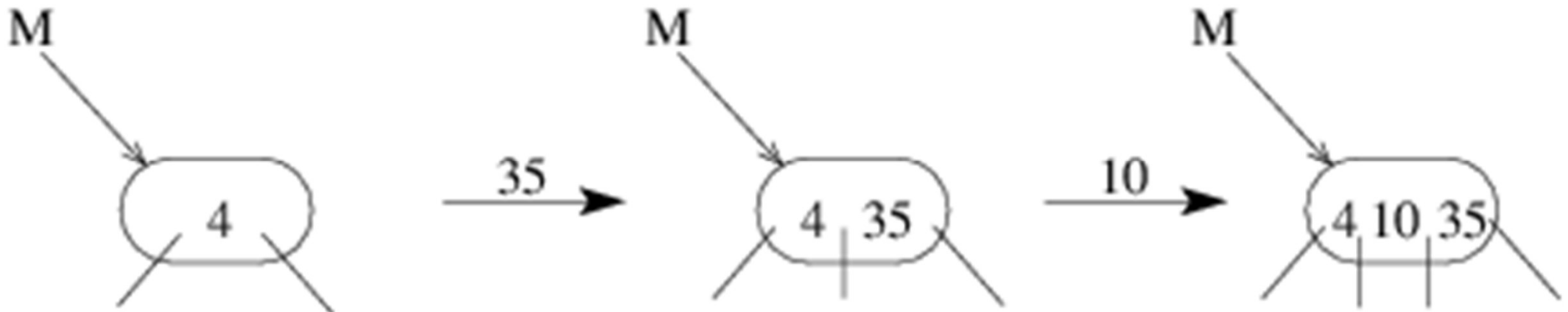
# Inserção com Fragmentação na Ascensão

- Inserimos os três primeiros elementos na raiz que é uma folha
- **Exemplificação (1): Faça a inserção do 4, 35 e 10**

Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

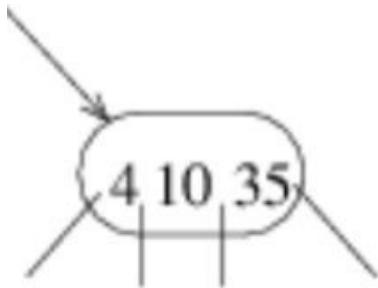
- Inserimos os três primeiros elementos na raiz que é uma folha



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

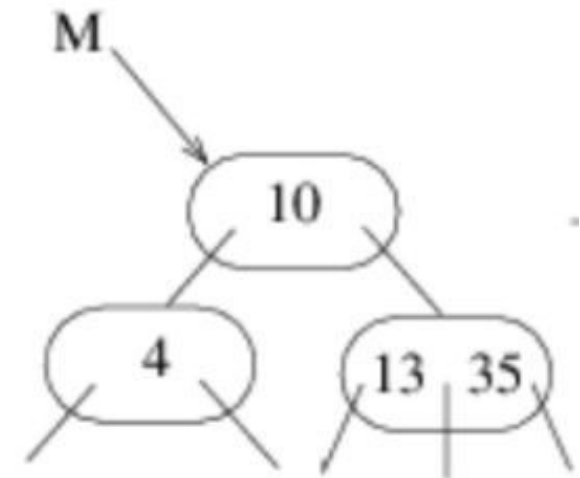
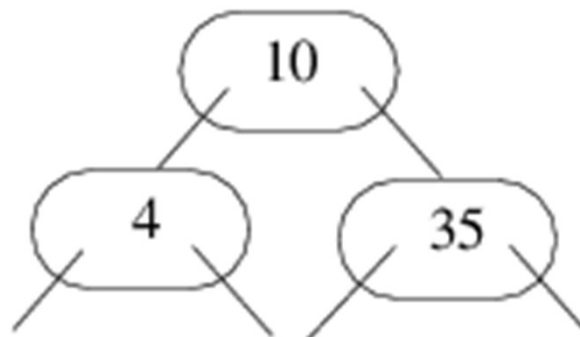
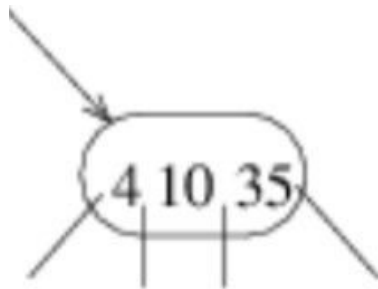
- Para o quarto elemento, fragmentamos a folha (4-nó) em três 2-nós onde o elemento do meio é pai dos demais e o novo elemento é inserido em uma das duas folhas
- **Exemplificação (2): Faça a inserção do 13**



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

- Para o quarto elemento, fragmentamos a folha (4-nó) em três 2-nós onde o elemento do meio é pai dos demais e o novo elemento é inserido em uma das duas folhas

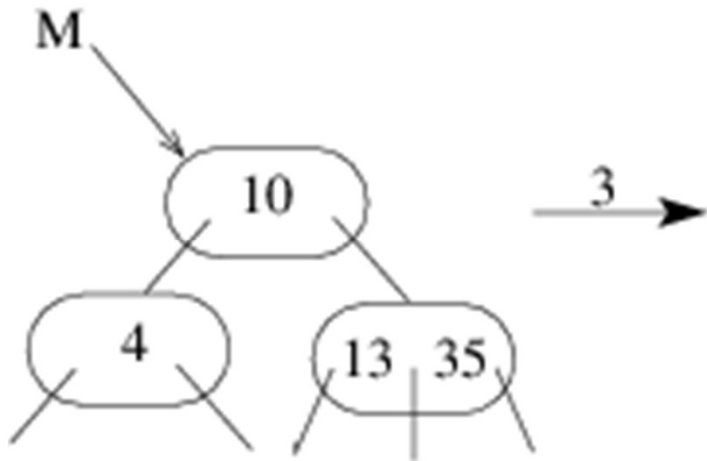


Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

• Para os demais elementos, fazemos uma pesquisa e os inserimos naturalmente nas folhas do tipo 2-nó ou 3-nó

- **Exemplificação (3): Faça a inserção do 3 e 30**



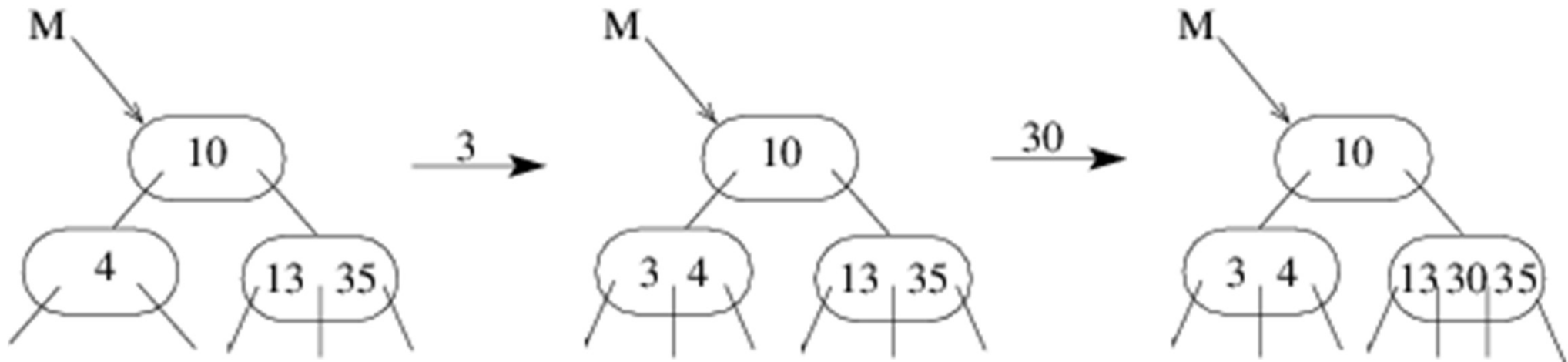
Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6



# Inserção com Fragmentação na Ascensão

• Para os demais elementos, fazemos uma pesquisa e os inserimos naturalmente nas folhas do tipo 2-nó ou 3-nó

• **Exemplificação (3): Faça a inserção do 3 e 30**

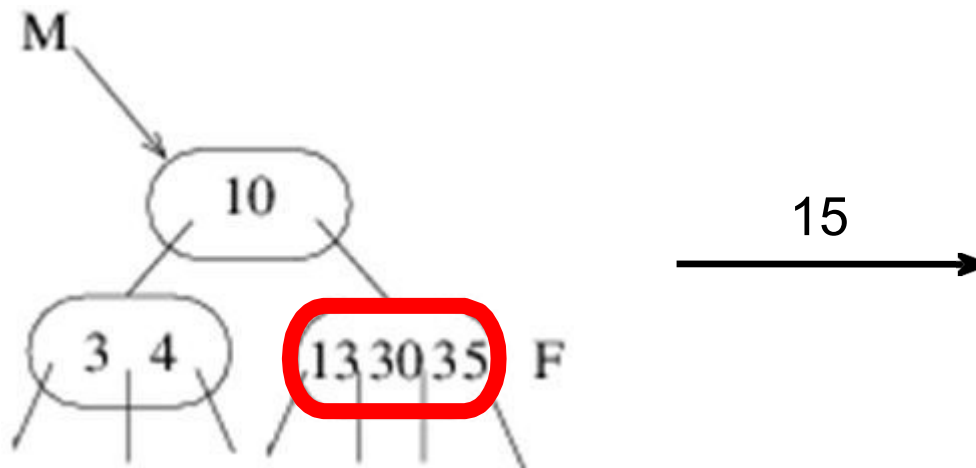


Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

- Quando a folha é um 4-nó, fragmentamos a mesma e efetuamos a ascensão do seu elemento do meio para o pai. O novo elemento ficará em uma das duas folhas resultantes da fragmentação

- **Exemplificação (4): Faça a inserção do 15**

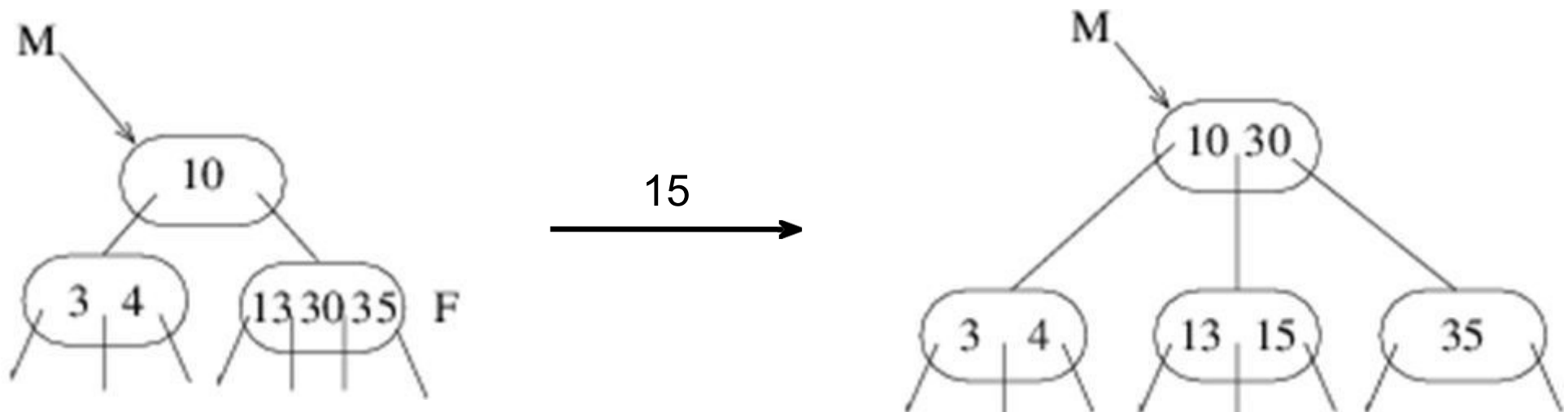


Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

- Quando a folha é um 4-nó, fragmentamos a mesma e efetuamos a ascensão do seu elemento do meio para o pai. O novo elemento ficará em uma das duas folhas resultantes da fragmentação

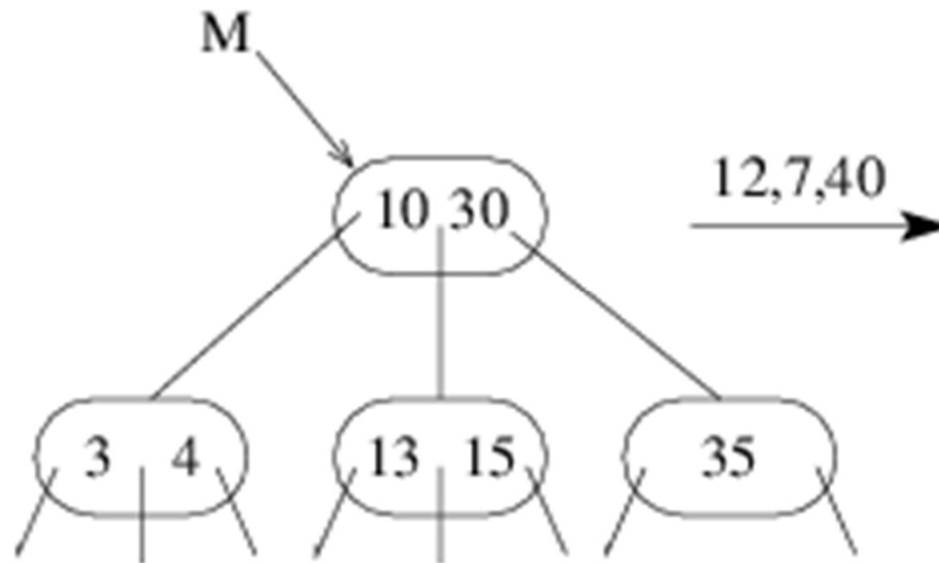
- **Exemplificação (4): Faça a inserção do 15**



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

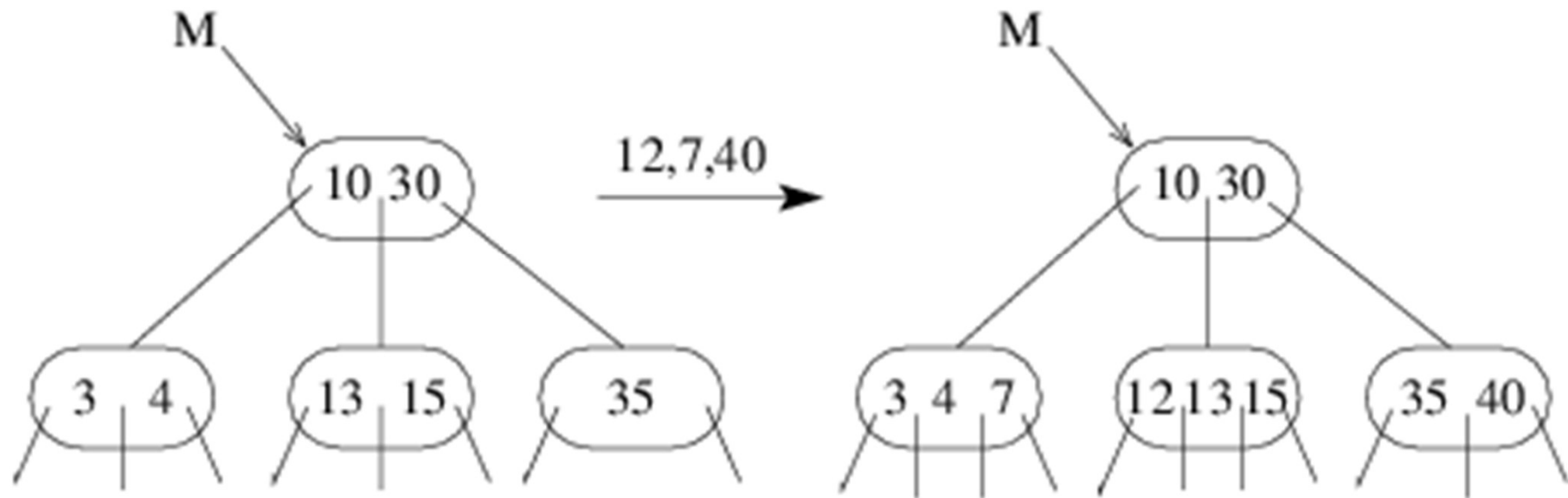
- **Exemplificação (5):** Faça a inserção do 12, 7 e 40



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

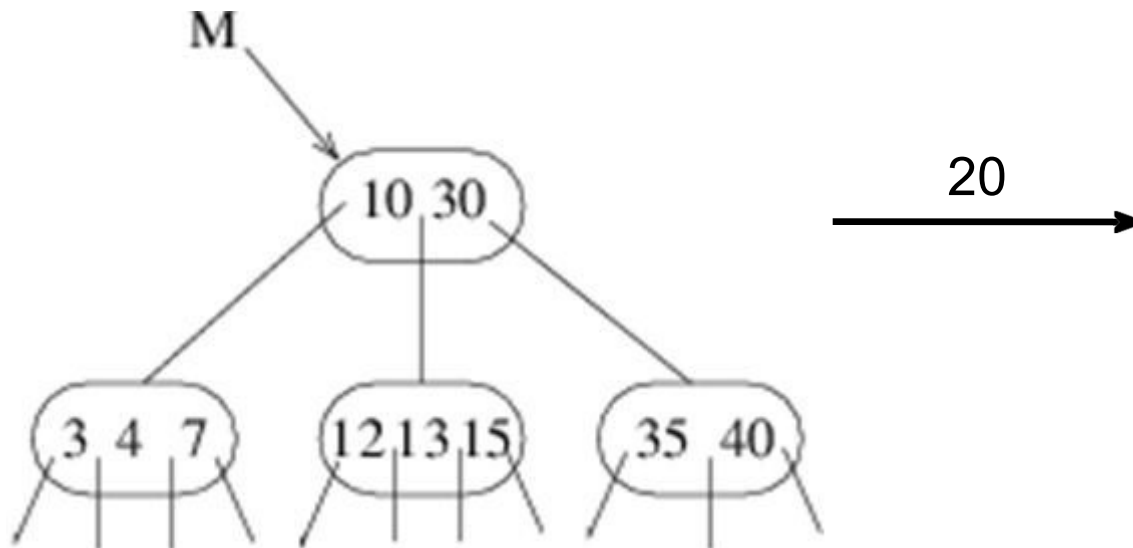
- Exemplificação (5): Faça a inserção do 12, 7 e 40



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

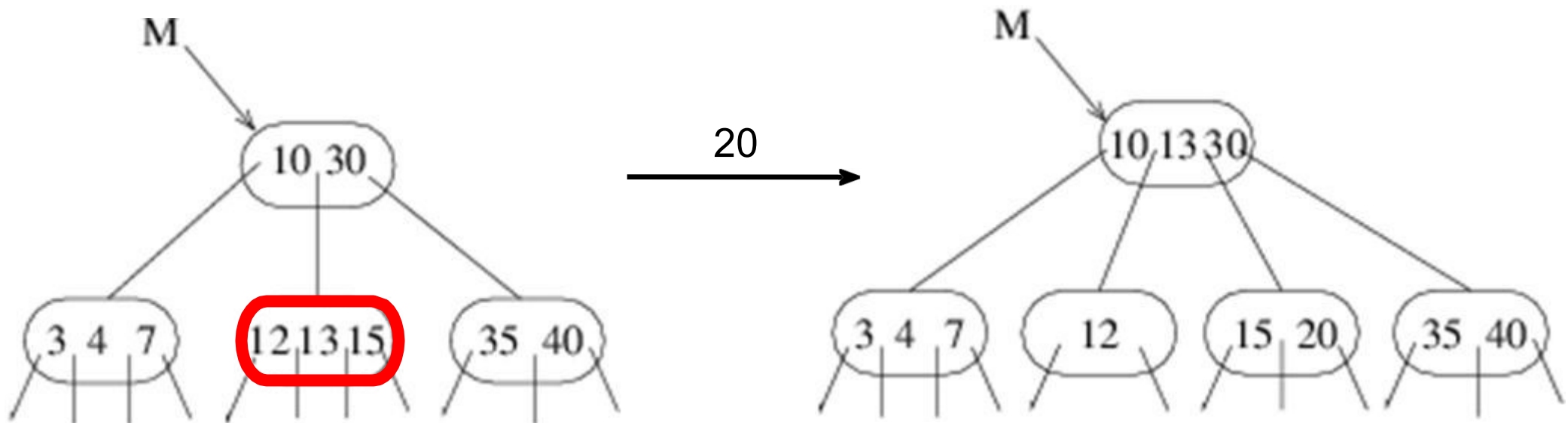
- Exemplificação (6): Faça a inserção do 20



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

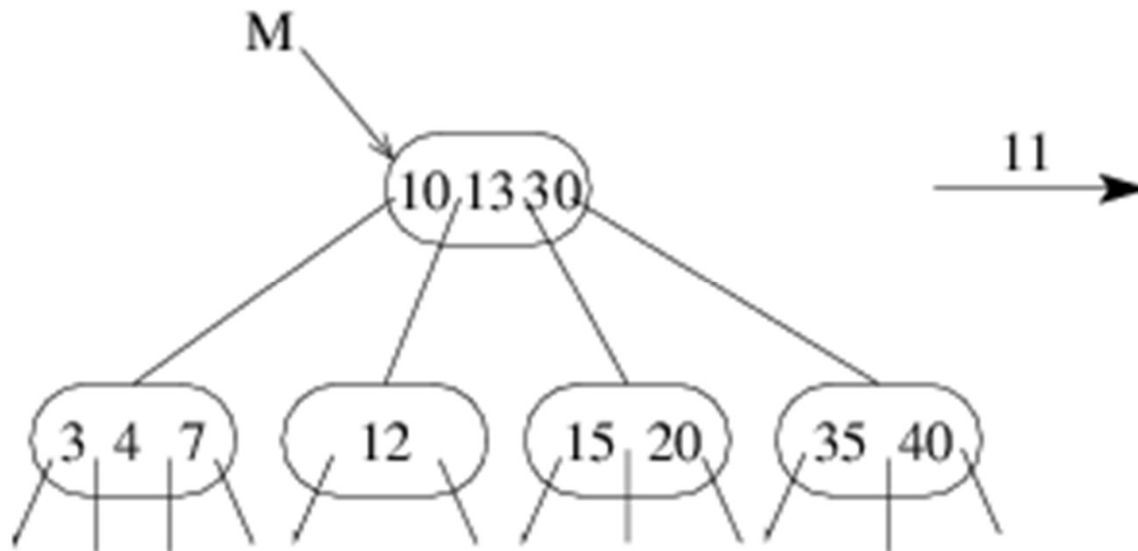
- **Exemplificação (6): Faça a inserção do 20.** Essa inserção causa a fragmentação do nó (12, 13, 15), criando um nó com (12), outro com o (15) e fazendo a ascensão do (13) para o nó (10, 30)



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão

- Exemplificação (7): Faça a inserção do 11

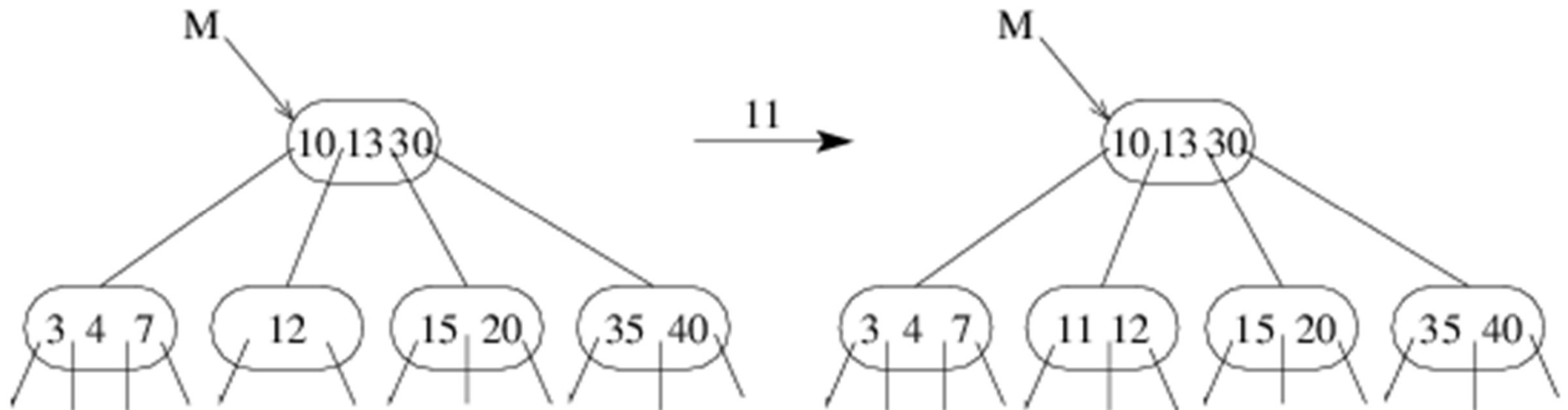


Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6



# Inserção com Fragmentação na Ascensão

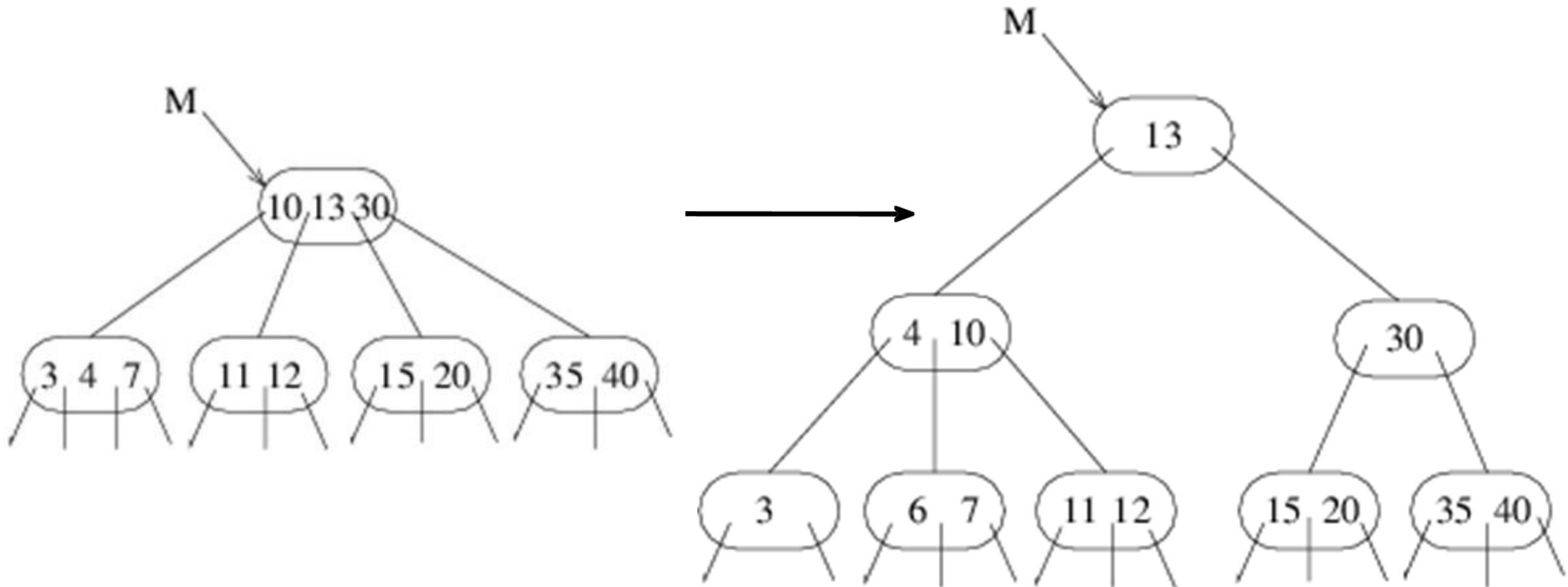
- Exemplificação (8): Faça a inserção do 11



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Ascensão


- **Exemplificação (8): Faça a inserção do 6.** Nesse caso, fragmentamos a folha (3,4,7) cujo pai é um 4-nó e que também é fragmentado, criando uma nova raiz



Elementos: **4**, **35**, **10**, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Cascata de Fragmentações

- Efeito colateral da fragmentação ascendente
- Acontece quando o caminho da inserção na árvore é formado somente por 4-nós, gerando fragmentações em toda a altura da árvore

- Introdução e Pesquisa
- Inserção com fragmentação por ascensão
- **Inserção com fragmentação na descida** 

# Eliminação da Cascata de Fragmentações

- Pode ser obtida proibindo que a árvore tenha dois 4-nós seguidos
- Garante que qualquer inserção causa, no máximo, uma fragmentação consecutiva
- Pode ser implementada fragmentando 4-nós na descida
  - Nesse caso, na pesquisa pela posição de inserção, chegando em um nó, primeiro, verificamos se o nó é um 4-nó. Em caso, positivo, fragmentamos o nó

# Fragmentação na Descida

- Técnica preventiva dado que qualquer 4-nó será fragmentado antes de qualquer inserção
- Uma desvantagem são as fragmentações “inúteis”

## Exemplificação (9)

- **O que acontece quando o pai de um nó é um: a) 2-nó; b) 3-nó; c) 4-nó?**
- O pai nunca será um 4-nó, pois, nesse caso, o pai seria anteriormente fragmentado

# Inserção com Fragmentação na Descida

- Apresentaremos esta técnica fazendo, como exemplo, a inserção dos números 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6



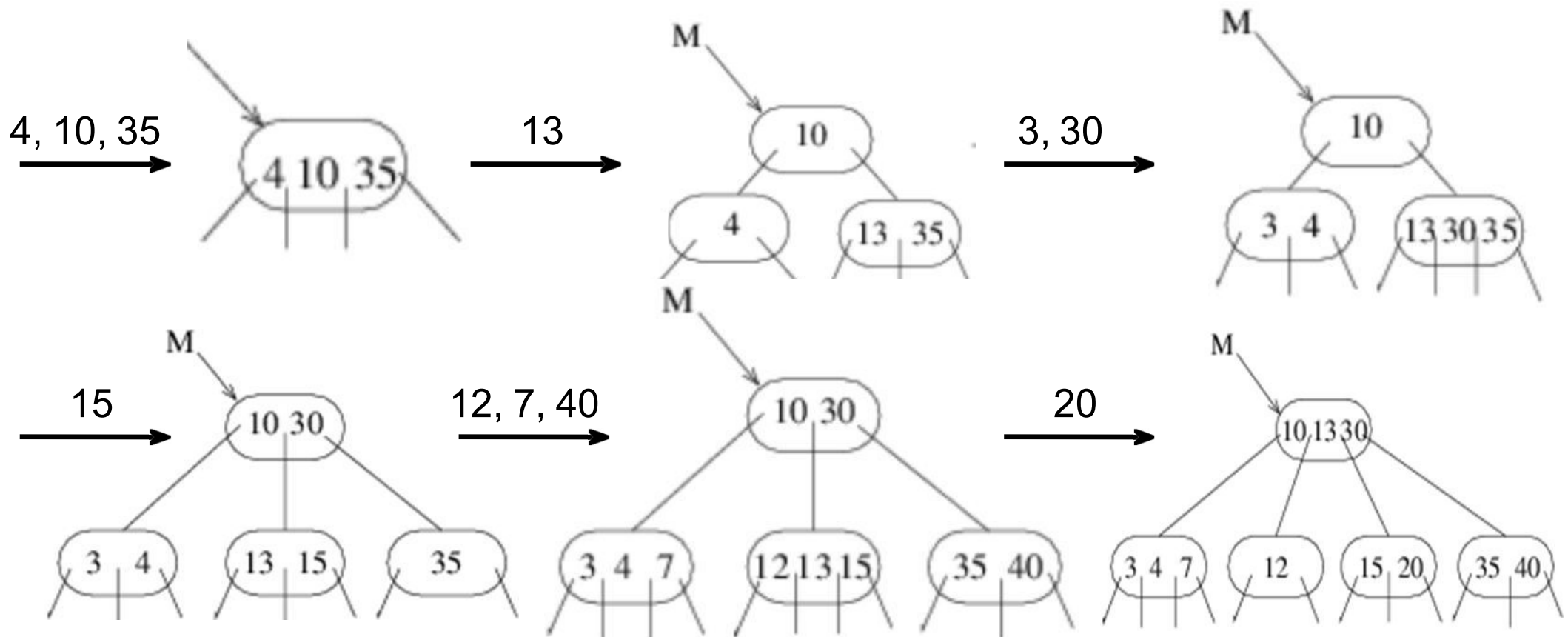
# Inserção com Fragmentação na Descida

- **Exemplificação (10):** Faça a inserção do 4, 35, 10, 13, 3, 30, 15, 12, 7, 40 e 20. Veja que as duas técnicas são iguais até este ponto

Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Descida

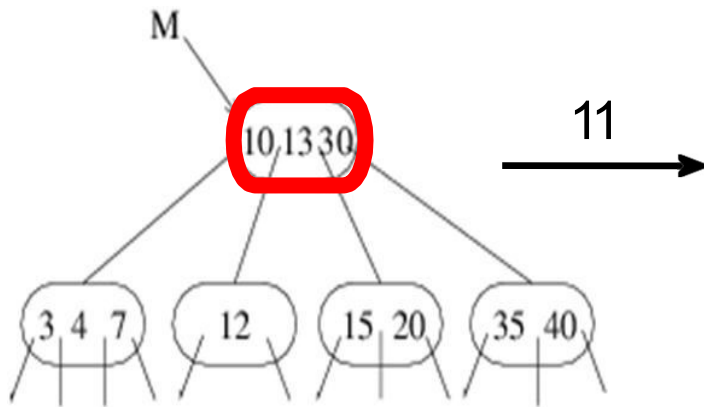
- Exemplificação (10):** Faça a inserção do 4, 35, 10, 13, 3, 30, 15, 12, 7, 40 e 20. Veja que as duas técnicas são iguais até este ponto



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Descida

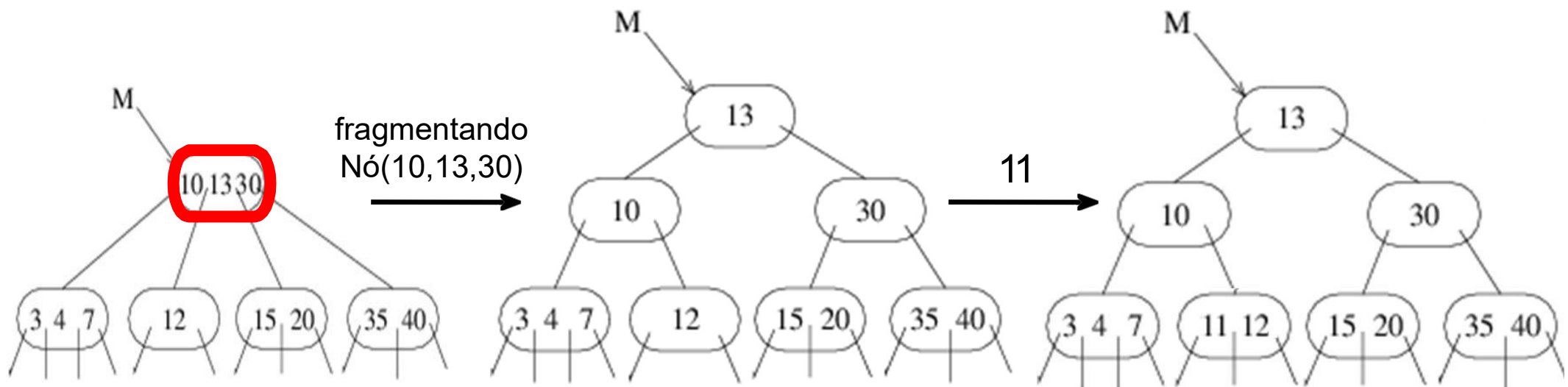
- **Exemplificação (11):** Faça a inserção do 11



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Descida

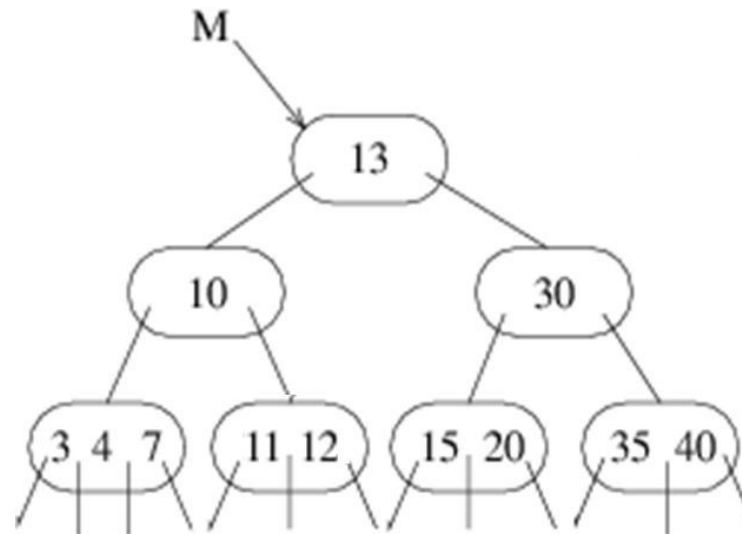
- Exemplificação (11): Faça a inserção do 11



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Descida

- **Exemplificação (12):** Neste ponto, qual é a diferença entre as árvores geradas pelas duas técnicas?

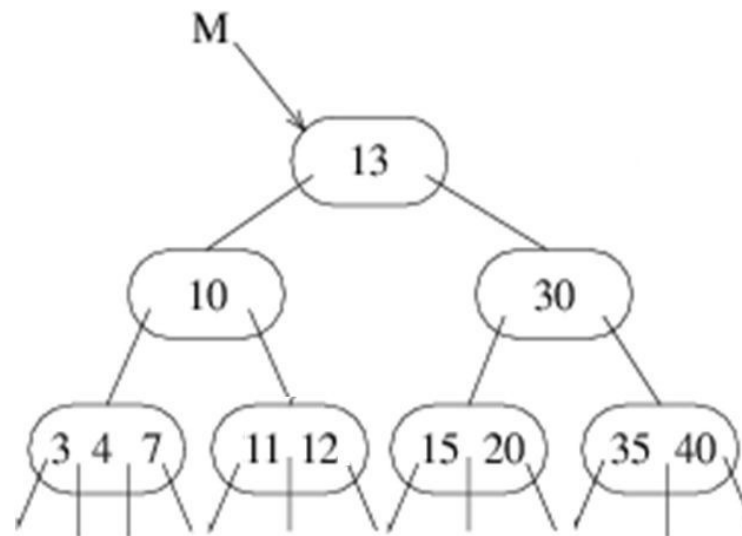


Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Descida

- **Exemplificação (12):** Neste ponto, qual é a diferença entre as árvores geradas pelas duas técnicas?

**Resposta:** Esta árvore tem um nível a mais

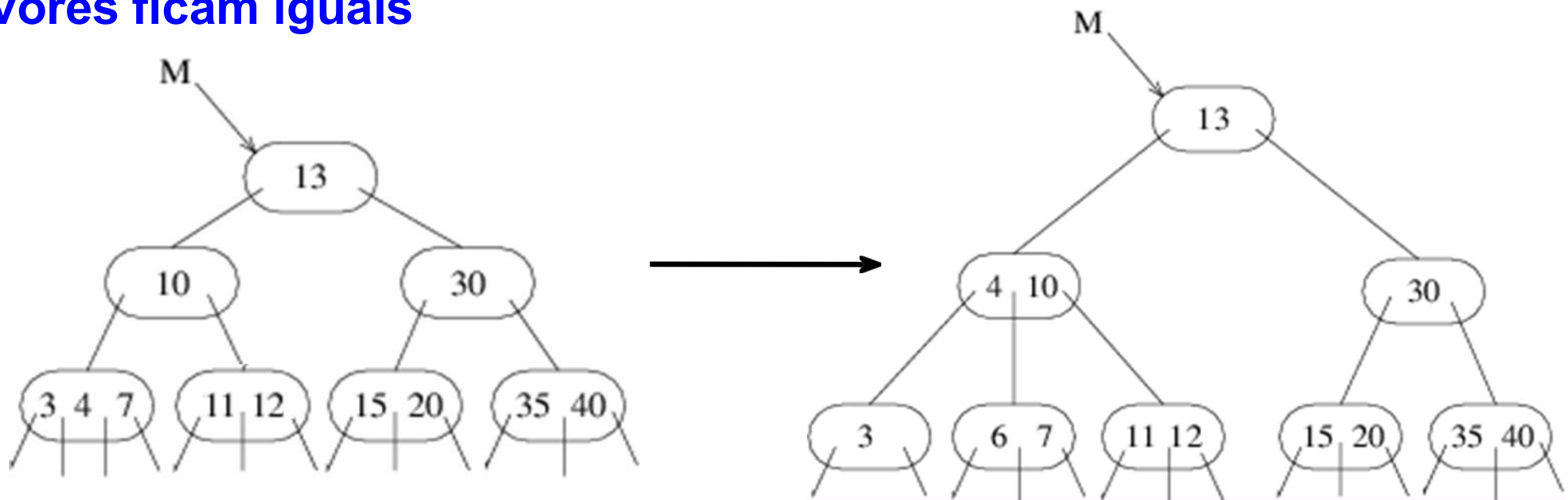


Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

# Inserção com Fragmentação na Descida

- **Exemplificação (12):** Neste ponto, qual é a diferença entre as árvores geradas pelas duas técnicas?

**Resposta:** Contudo, após a inserção do 6, observamos que as duas árvores ficam iguais



Elementos: 4, 35, 10, 13, 3, 30, 15, 12, 7, 40, 20, 11 e 6

## Observações

- Nas duas técnicas, a altura da árvore aumenta quando fragmentamos a raiz
- A segunda técnica faz todos seus reequilíbrios durante a descida na árvore
- Na segunda técnica, os reequilíbrios são puramente locais
- As duas técnicas sempre operam em um caminho da raiz até uma folha
- A complexidade para o número de comparações das duas técnicas no pior e no caso médio é  $\Theta(\lg(n))$



# Observações

- A primeira técnica demanda uma pilha para restaurar o equilíbrio da árvore, repassando o caminho inverso de pesquisa
- A segunda técnica é mais fácil de paralelizar dado que os reequilíbrios são locais e não existe a dependência do caminho inverso de pesquisa
- Uma desvantagem da segunda técnica é que ela pode consumir mais memória, pois sua taxa de ocupação dos nós é menor do que a outra

# Técnica Pró-ativa vs. Reativa

- A inserção com fragmentação por ascensão é reativa, a na descida, pró-ativa
- Técnicas reativas esperam um problema para atuar
- Técnicas pró-ativas se adiantam e tentam resolver/minimizar o problema antes que o mesmo aconteça
- Várias soluções na Computação são classificadas como pró-ativas e reativas

# Próxima aula

Tabela *hash*