



# Conteúdo

1. Introdução
2. Listas
3. Pilhas e Filas
4. Árvores
5. Árvores de Pesquisa
  - Árvore Binária e Árvore AVL
  - **Árvore N-ária** e Árvore B
6. Tabelas de Dispersão (Hashing)
7. Métodos de Acesso a Arquivos
8. Métodos de Ordenação de Dados





# Árvores N-árias de Pesquisa



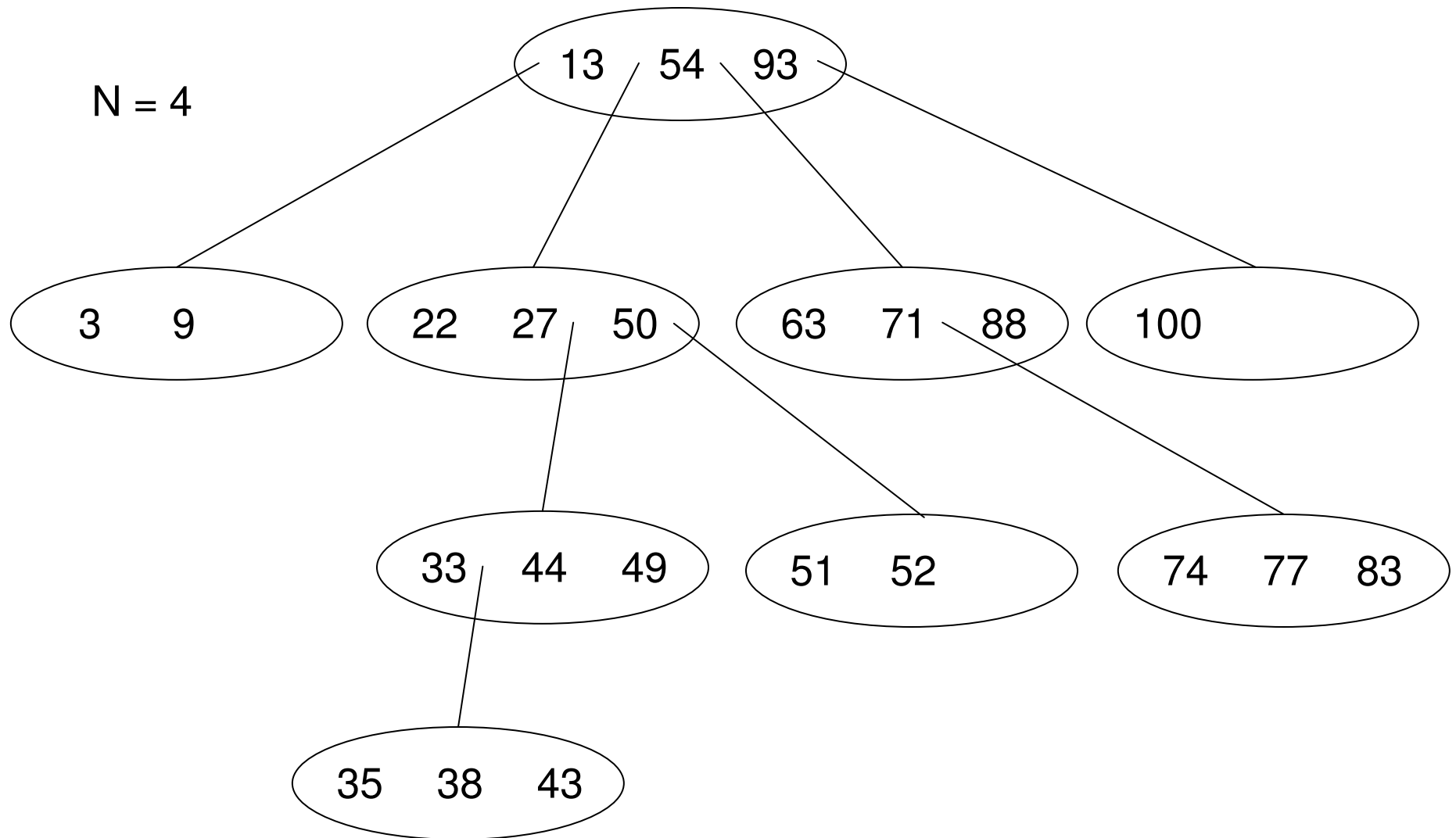
# Árvore N-ária de Pesquisa

Uma **Árvore N-ária de Pesquisa** ou **Árvore N-ária de Busca** é uma árvore que :

- contém  $\underline{n}$  subárvores e  $\underline{m}$  chaves, sendo  $\underline{m} = \underline{n} - 1$  e  $2 \leq \underline{n} \leq N$ ;
- todas as chaves estão ordenadas, ou seja, dado um conjunto de chaves  $ch_1, ch_2, \dots, ch_i, \dots, ch_m$ , nesta ordem, tem-se:  $ch_i < ch_{i+1}$

# Exemplo de Árvore N-ária de Pesquisa

N = 4

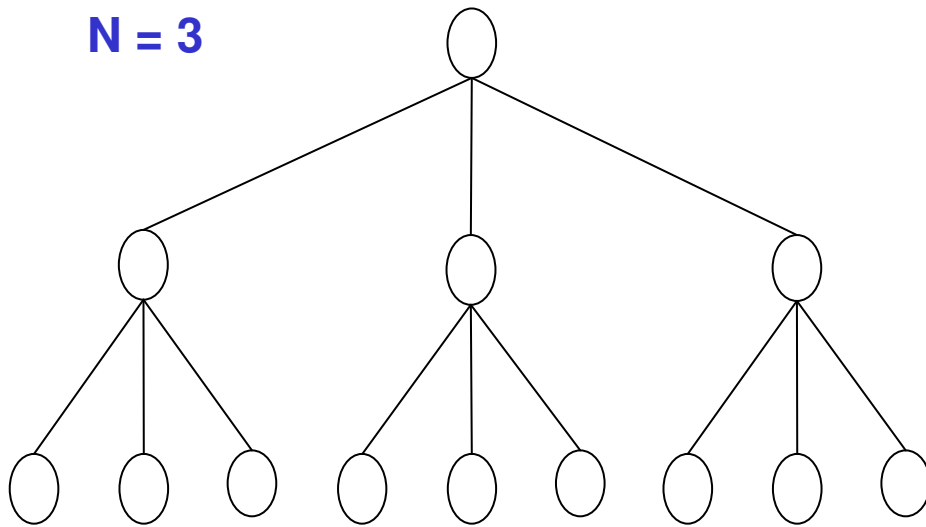


•  
•  
•

# Vantagem de uma Árvore N-ária de Pesquisa

Indexação de um grande volume de dados:

**N = 3**



**$h(A) = 0 \rightarrow 2$  chaves**

**$h(A) = 1 \rightarrow 2 + 3 \cdot 2 = 8$  chaves**

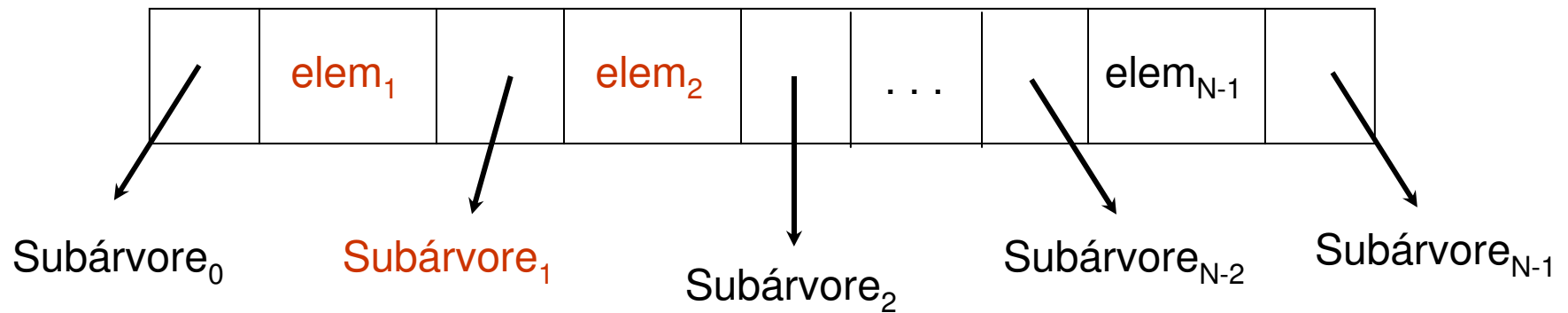
**$h(A) = 2 \rightarrow 2 + 3 \cdot 2 + 3 \cdot 3 \cdot 2 = 26$  chaves**

*“Quanto maior o N, maior é o número de chaves que se pode indexar e conseqüentemente, encontra-se uma chave com menos acessos à árvore”*



• • • • • • • • •

# Modelagem Física



⇒  $\text{Sub\acute{a}rvore}_i$  mantém todos os elementos maiores que  $\text{elemento}_i$  e menores que  $\text{elemento}_{i+1}$



# Implementação

numElem

6

elementos

*null*

12

39

51

77

89

102

subÁrvores

Sub<sub>0</sub>

Sub<sub>1</sub>

Sub<sub>2</sub>

Sub<sub>3</sub>

Sub<sub>4</sub>

Sub<sub>5</sub>

Sub<sub>6</sub>





# Operações em uma Árvore N-ária de Pesquisa

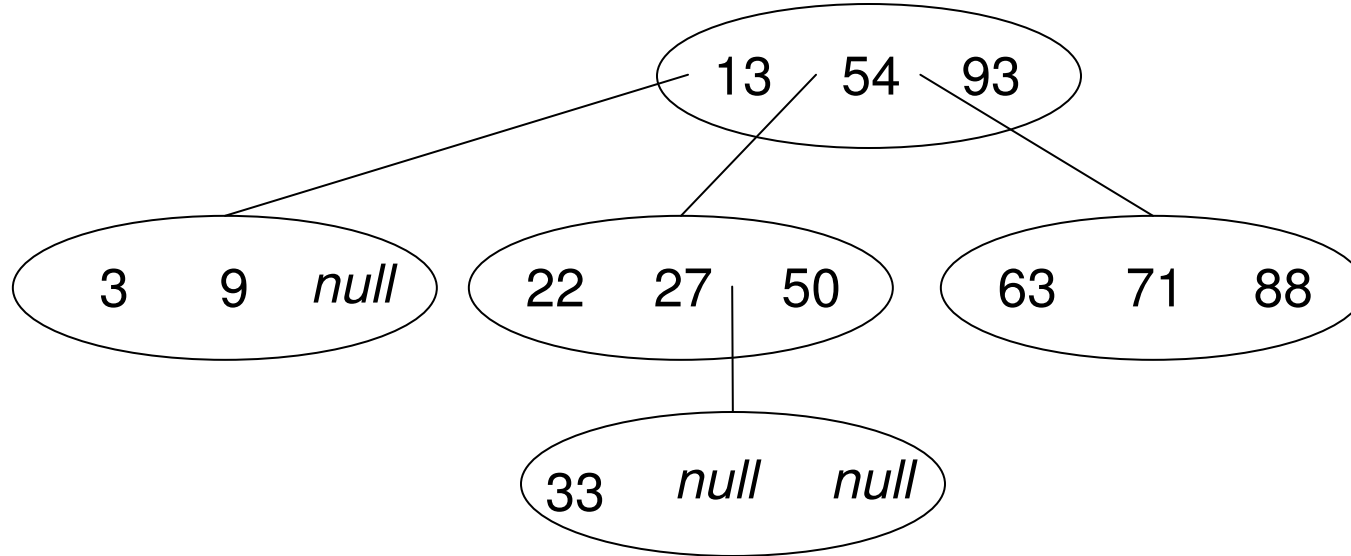
- Pesquisa
  - pesquisa todos os nodos
  - pesquisa um elemento
- Inclusão (elemento)
- Exclusão (elemento)





# Pesquisa de Todos os Elementos

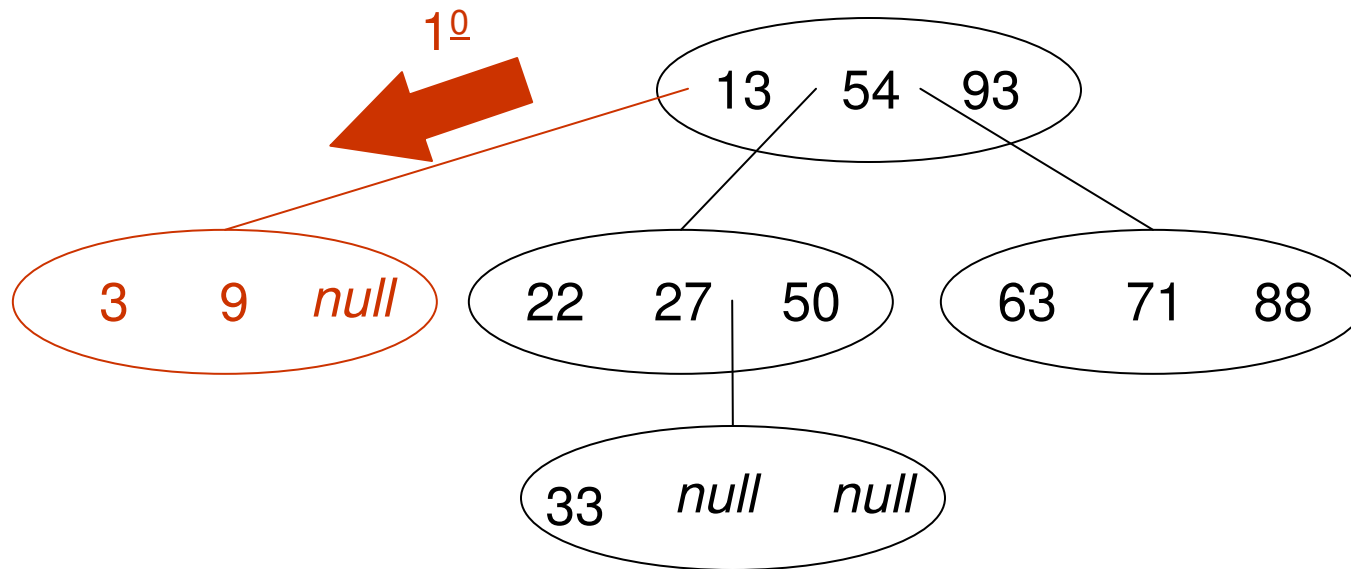
➡ Busca in-ordem





# Pesquisa de Todos os Elementos

➡ Busca in-ordem



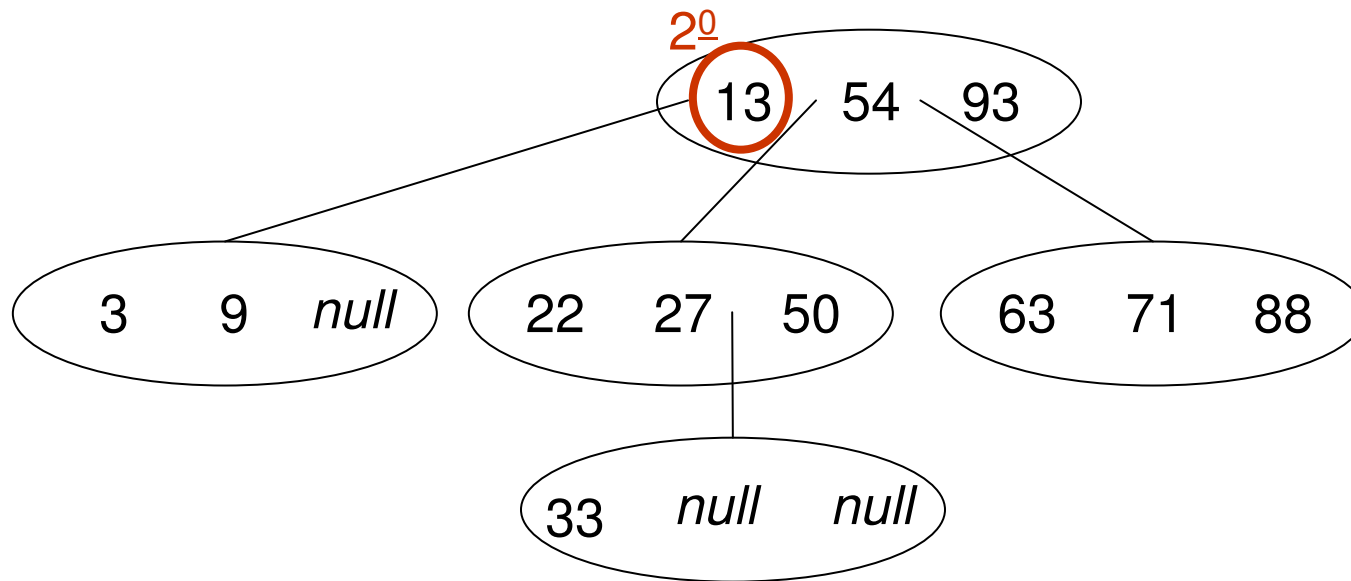
**3-9**



...

# Pesquisa de Todos os Elementos

➡ Busca in-ordem

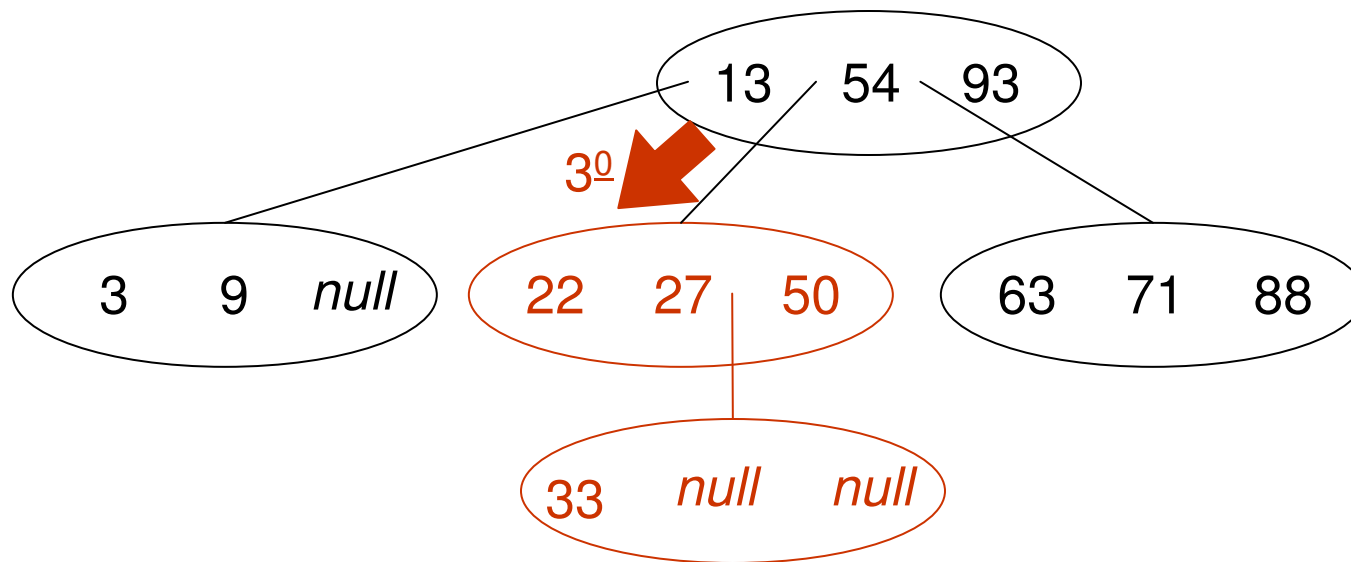


**3-9-13**

.....

# Pesquisa de Todos os Elementos

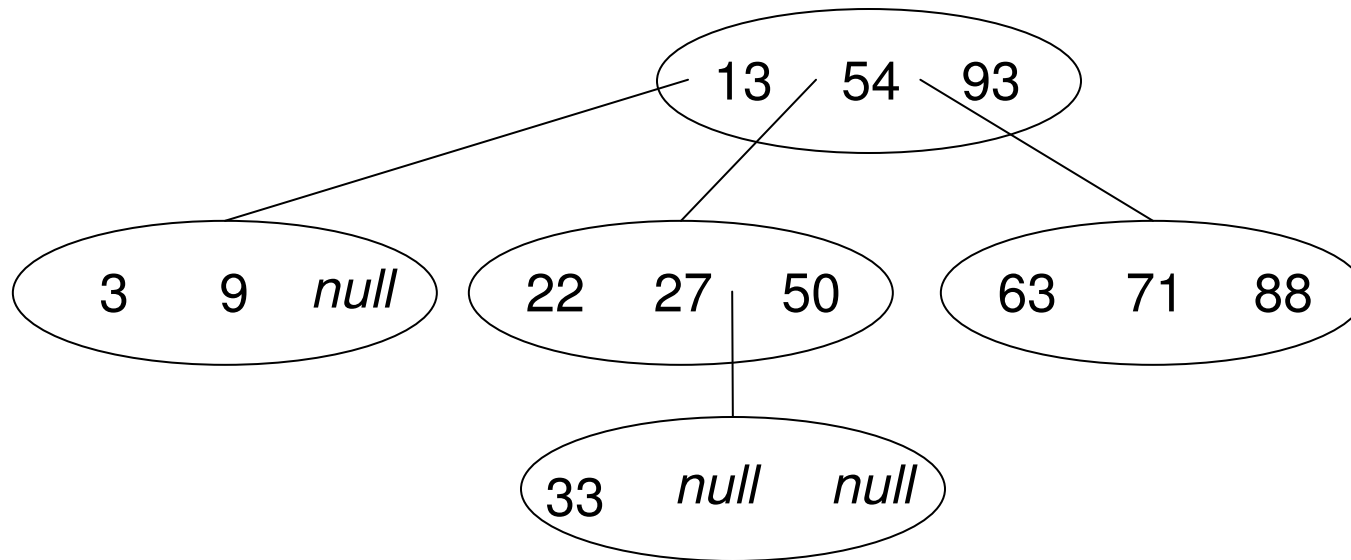
➡ Busca in-ordem



**3-9-13-22-27-33-50**

# Pesquisa de Todos os Elementos

➡ Busca in-ordem



**3-9-13-22-27-50-54-63-71-88-93**



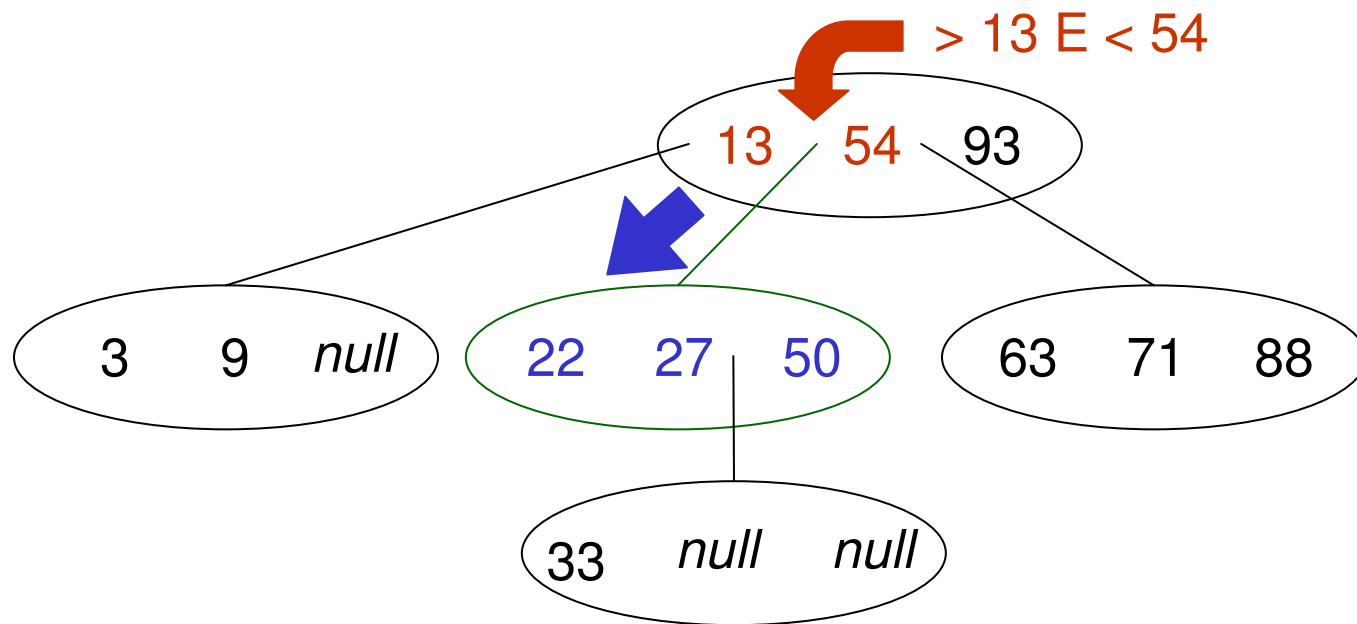
## Pesquisa de um Elemento

- Busca dos elementos dentro do nodo:
  - busca seqüencial
  - busca binária
- Se o elemento do nodo  $>$  elemento pesquisado  $\rightarrow$  árvore ESQ
- Se o elemento do nodo  $<$  elemento pesquisado  $\rightarrow$  árvore DIR



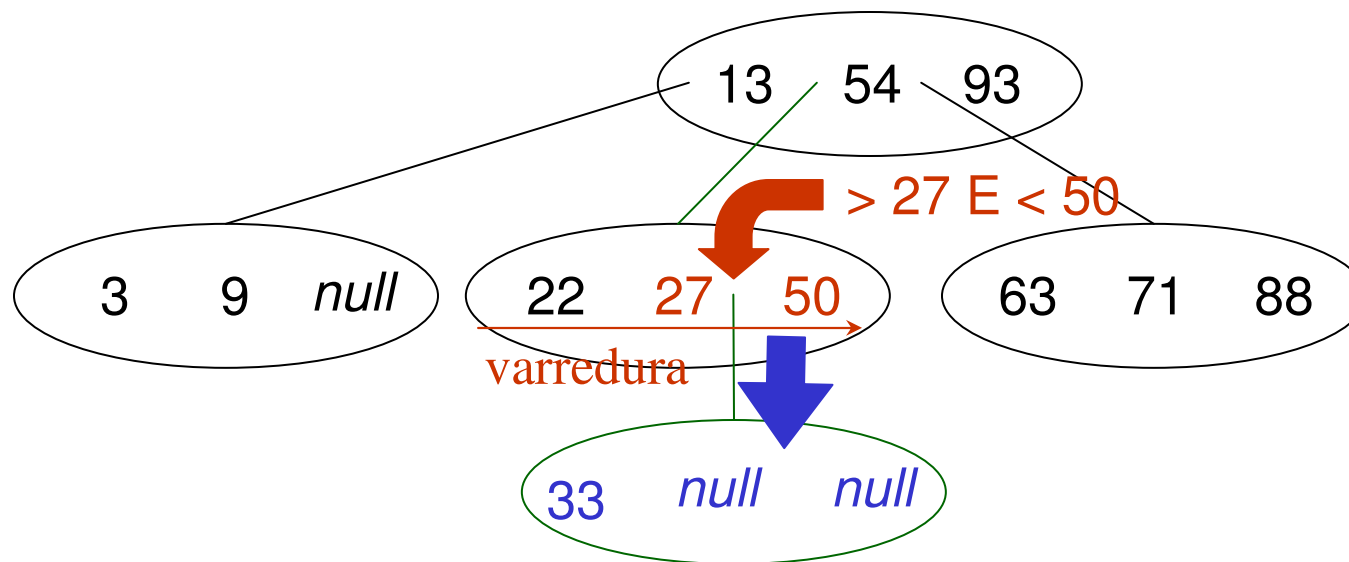
# Busca Seqüencial

Pesquisar o elemento 33.



# Busca Seqüencial

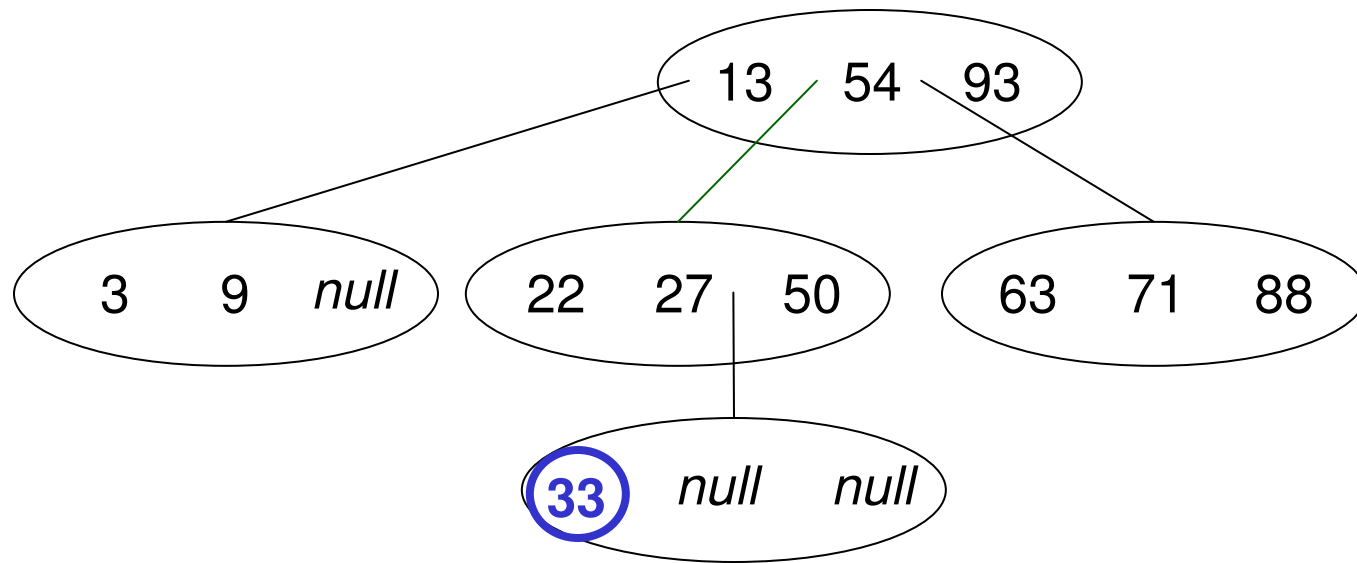
Pesquisar o elemento 33.





# Busca Seqüencial

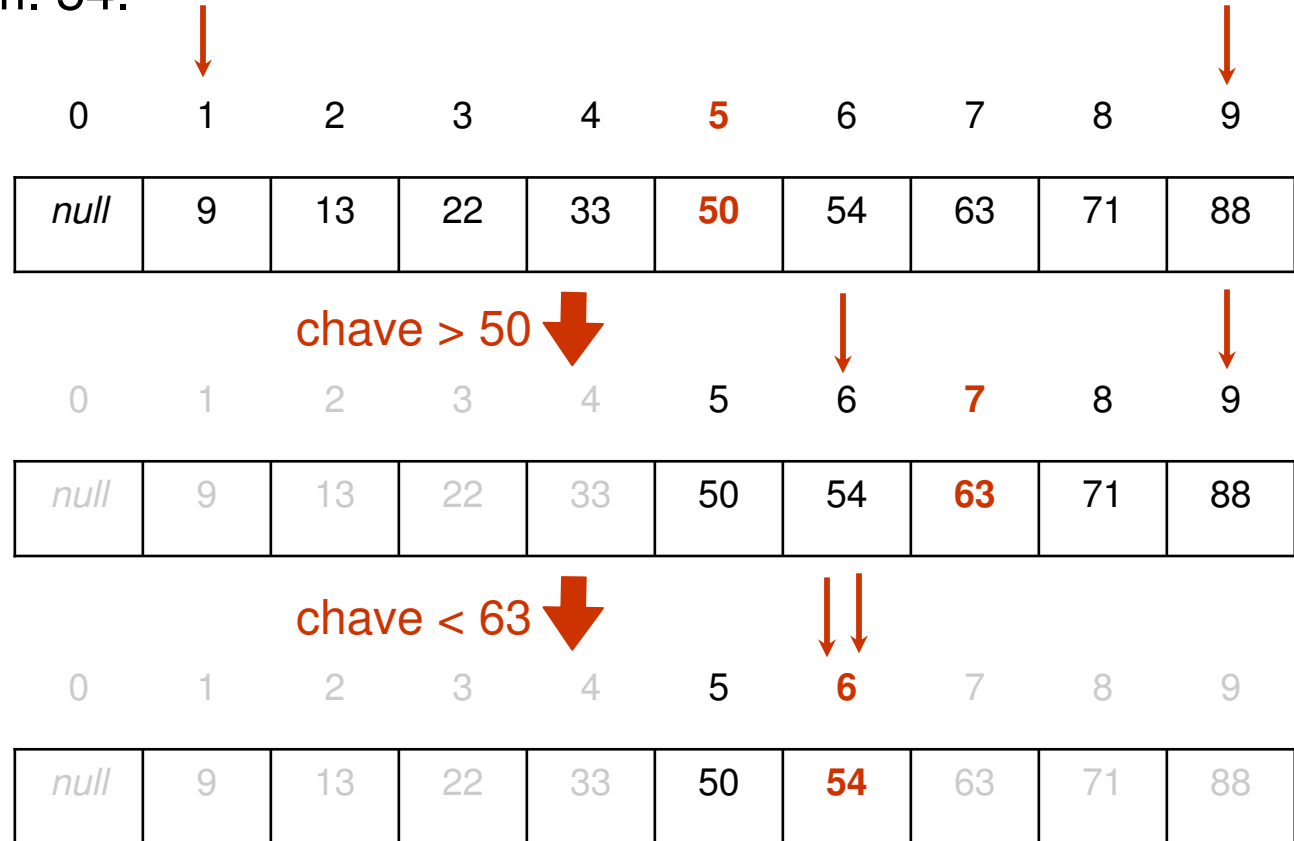
Pesquisar o elemento 33.



# Busca Binária

A varredura dos elementos de um nodo é feita através de uma pesquisa binária.

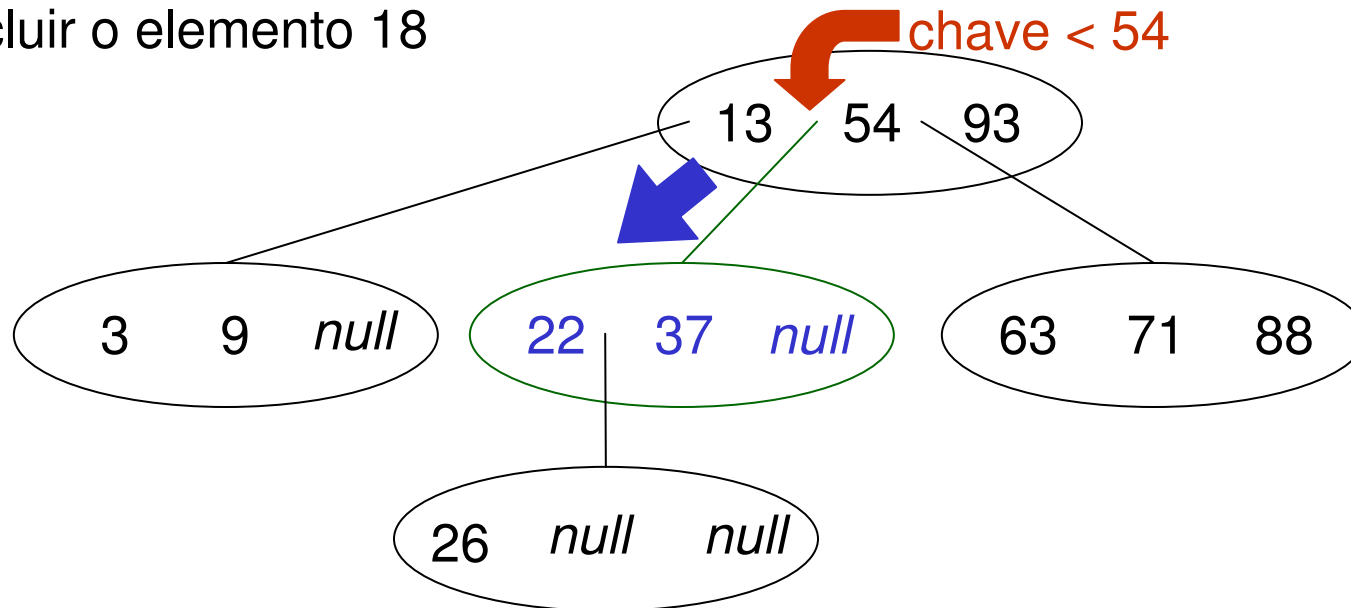
Pesquisar o elem. 54.



# Inclusão em uma Árvore N-ária de Pesquisa

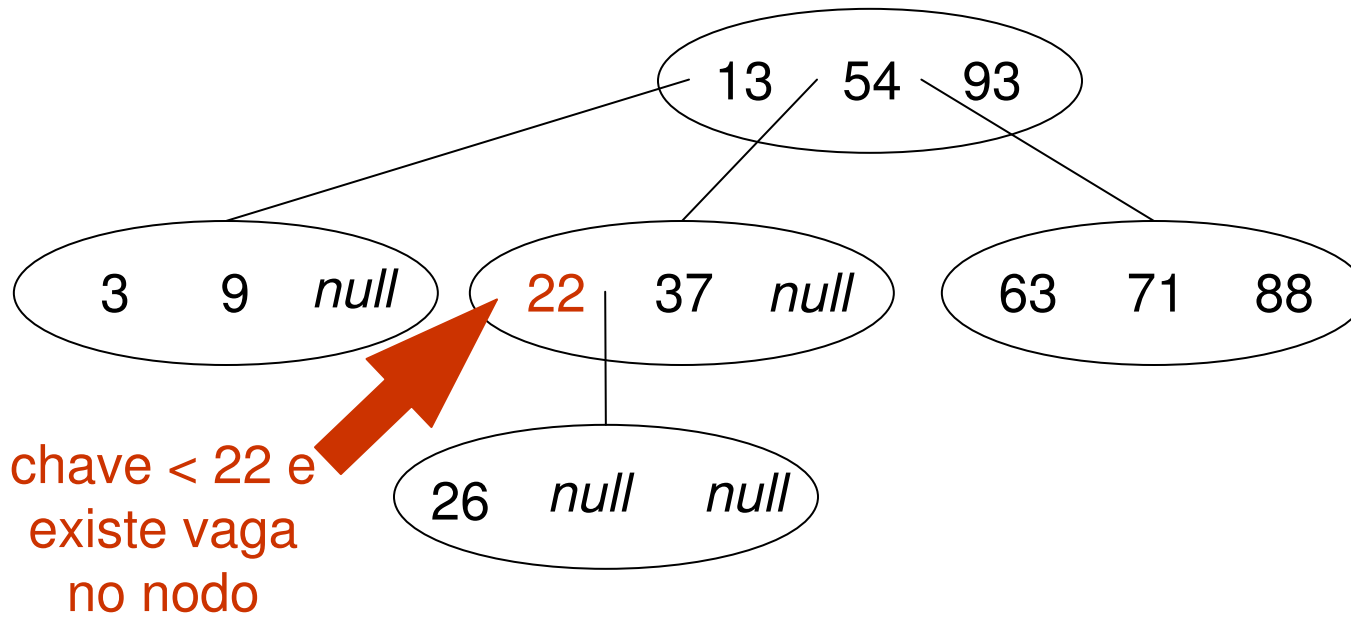
- Busca a posição na qual o elemento deve ser inserido.
- Caso haja espaço para inserir o elemento em um nodo, o vetor deve ser rearranjado (deslocamento de elementos e subárvores) para a sua correta colocação.

Incluir o elemento 18



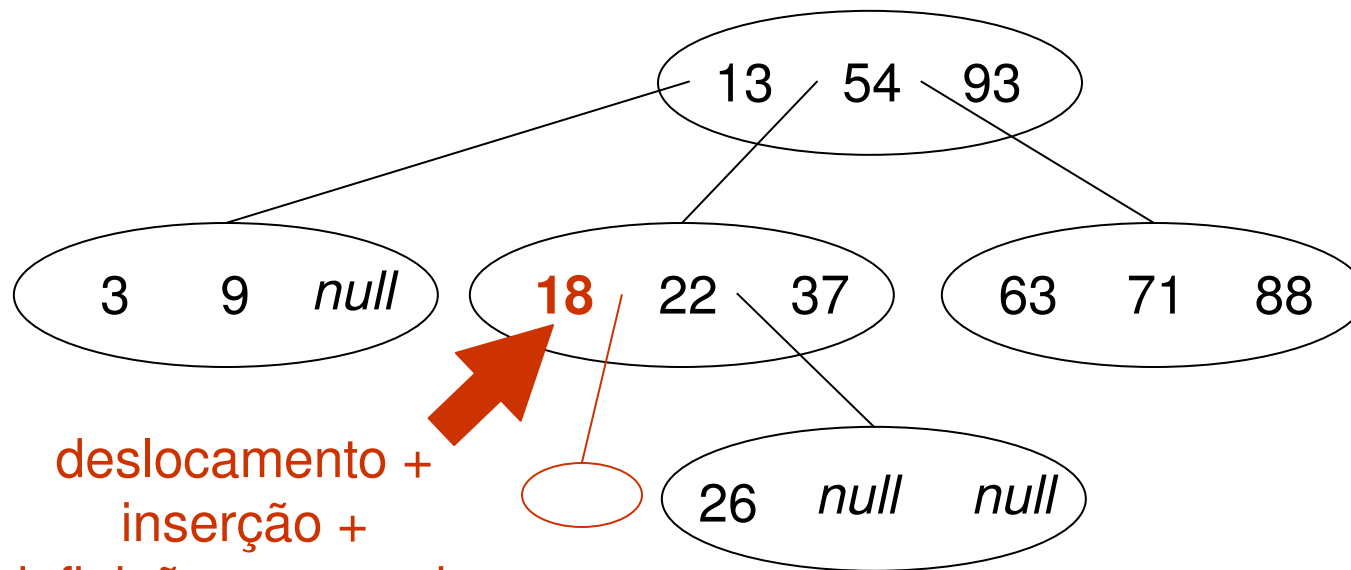
# Inclusão

Incluir o elemento 18



# Inclusão

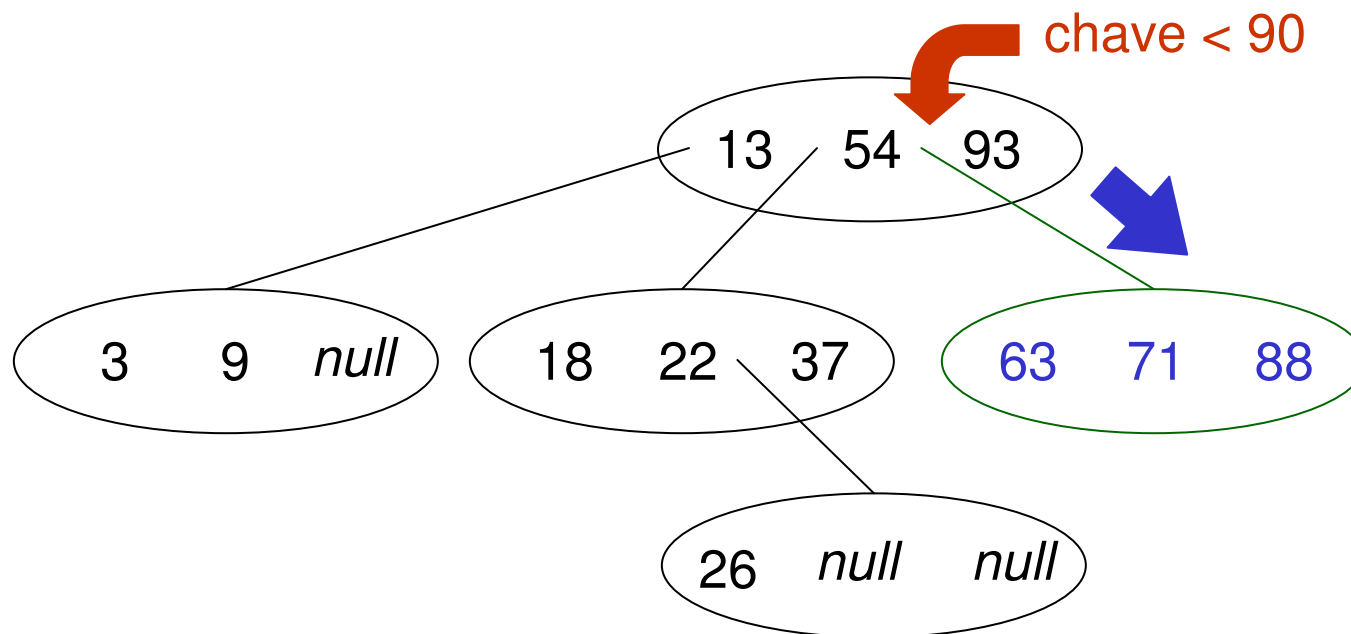
Incluir o elemento 18



deslocamento +  
inserção +  
definição novo nodo  
vazio à direita

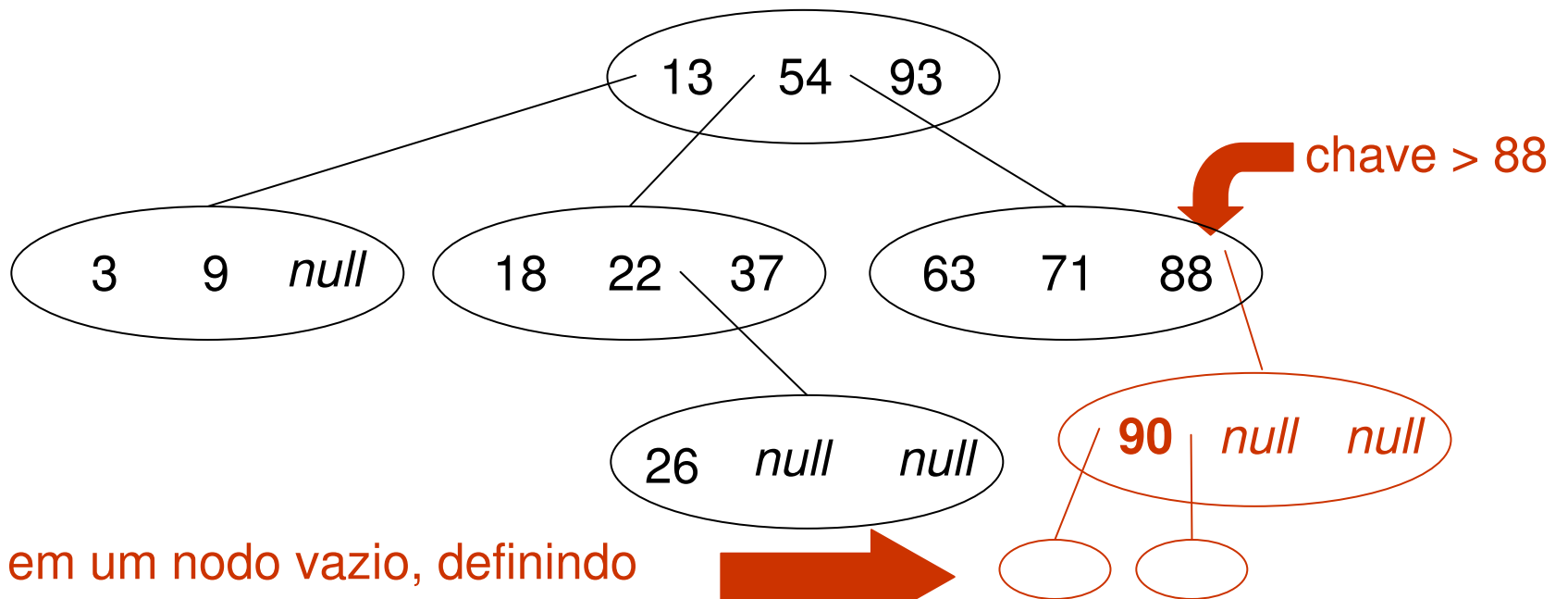
# Inclusão

Incluir o elemento 90



# Inclusão

Incluir o elemento 90



incluir em um nodo vazio, definindo  
uma subárvore à esquerda e à direita



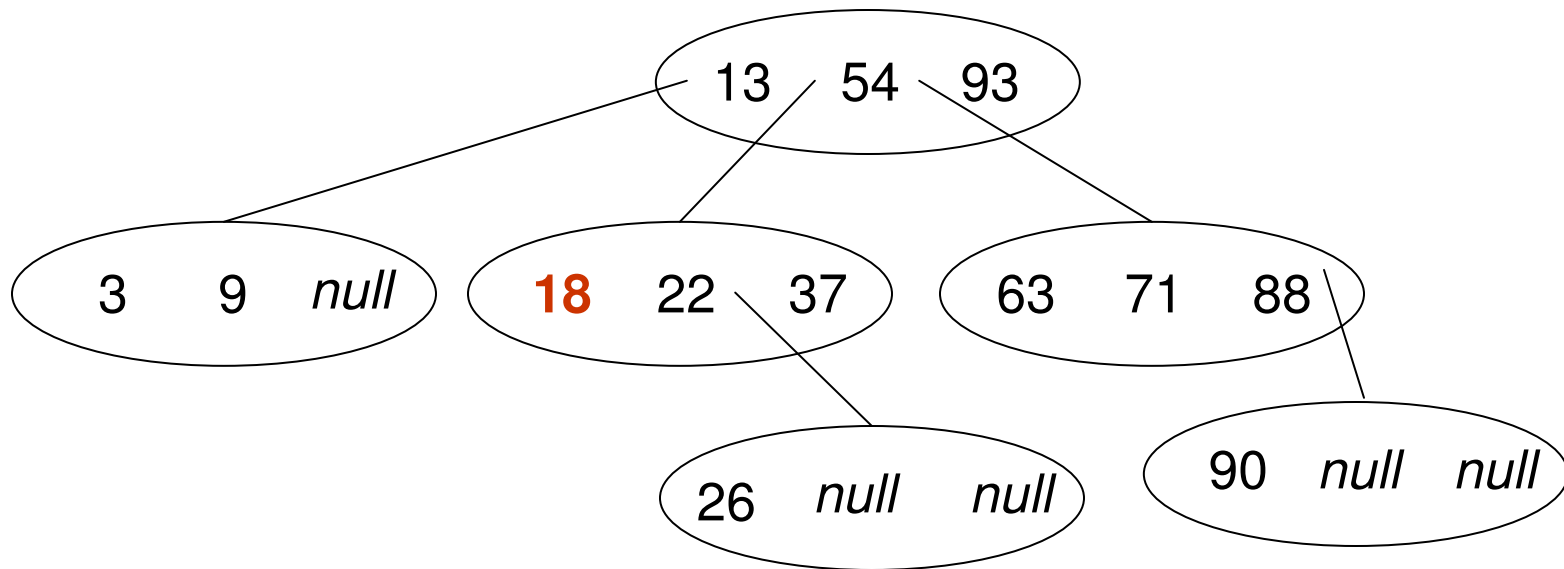
## Exclusão em uma Árvore N-ária de Pesquisa

- Se o elemento possui subárvores ESQ e DIR vazias, ele é removido e ocorre deslocamento no vetor para ajustar os elementos restantes.
- Se o elemento possui subárvores ESQ e DIR não vazias, ele é trocado com o maior elemento da subárvore ESQ ou o menor elemento da subárvore DIR (processo recursivo – até que ele esteja em um nodo folha!)



# Exclusão

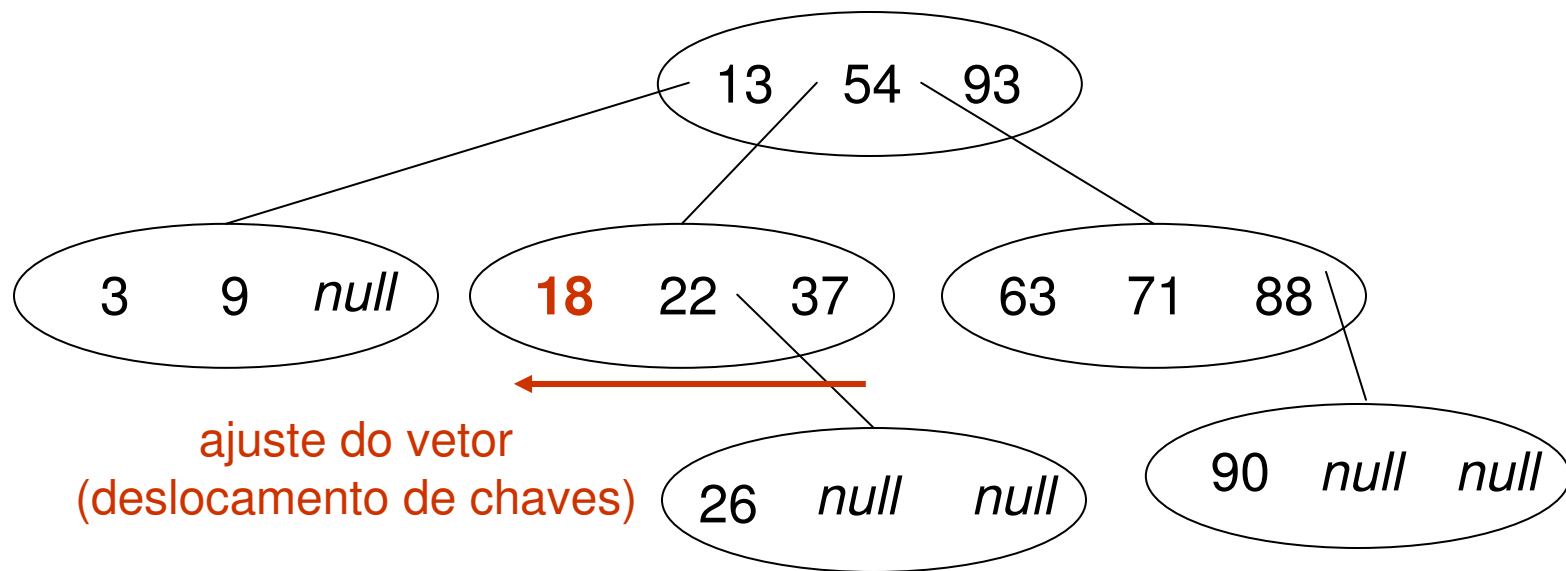
Excluir o elemento 18



# Exclusão

Excluir o elemento 18

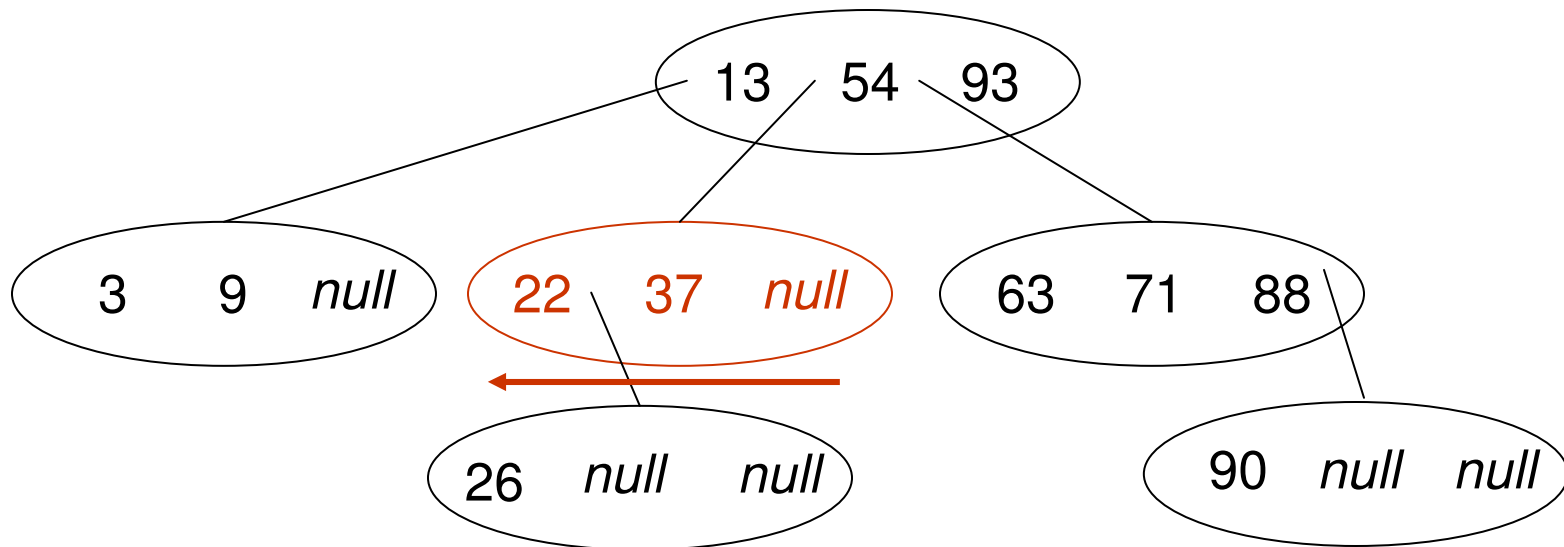
→ não há subárvores ESQ e DIR



# Exclusão

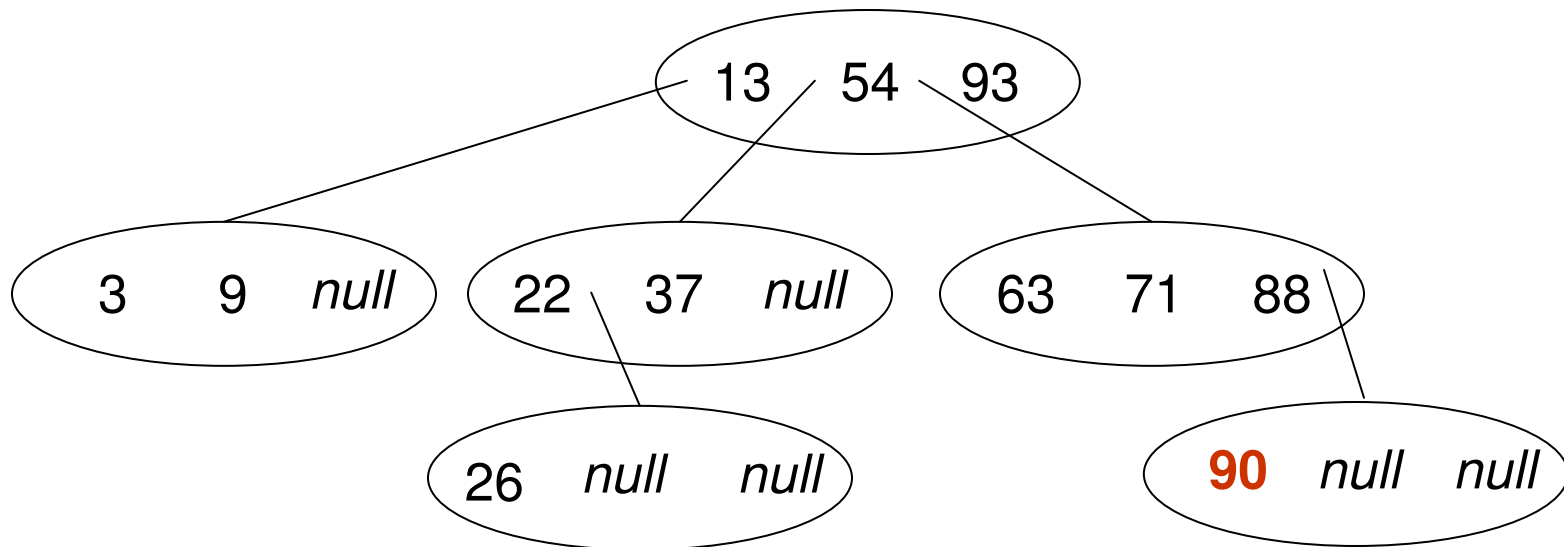
Excluir o elemento 18

→ não há subárvores ESQ e DIR



# Exclusão

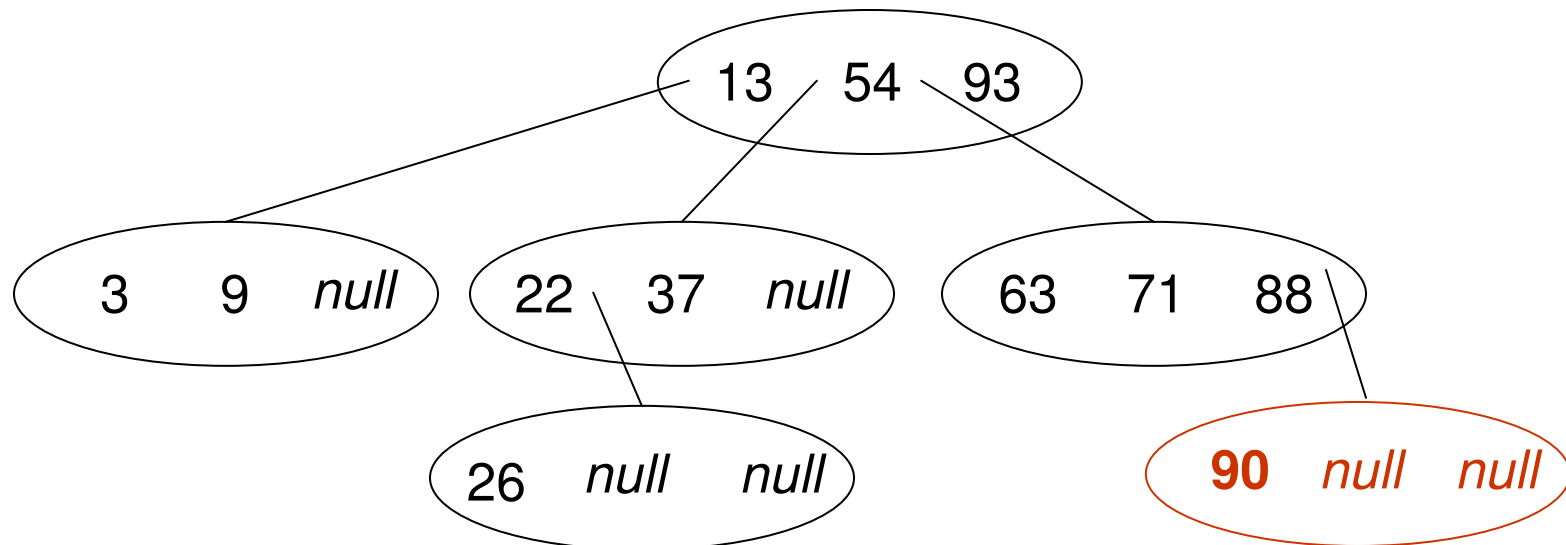
Excluir o elemento 90



# Exclusão

Excluir o elemento 90

→ não há subárvores ESQ e DIR e o elemento é o único do nodo

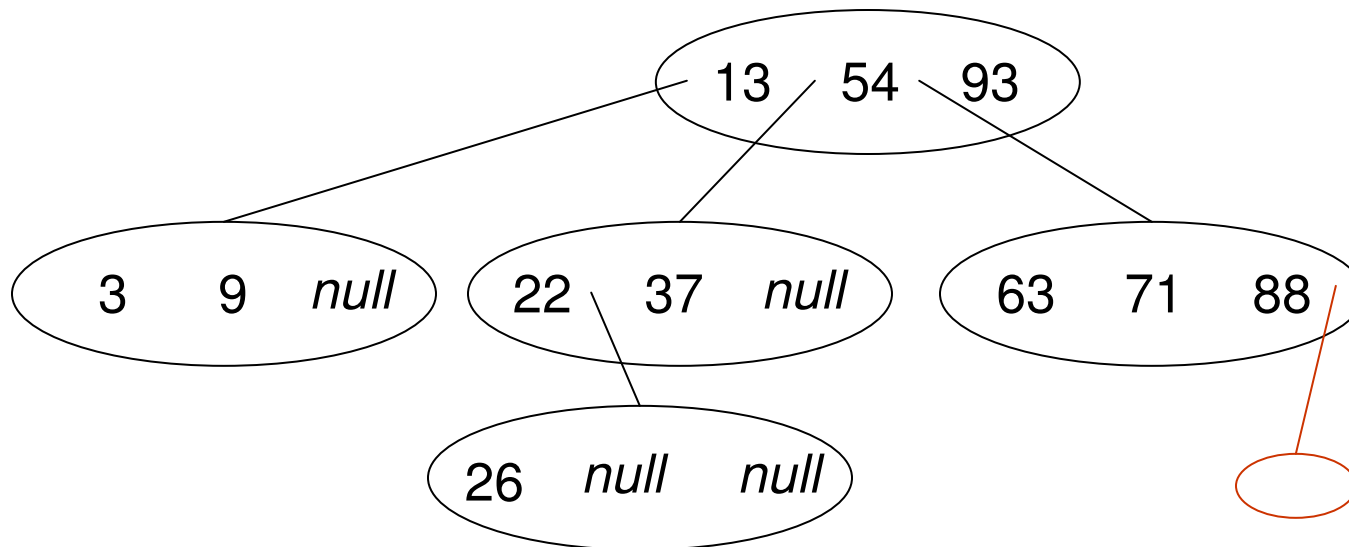


nodo torna-se vazio  
(subárvores ESQ e DIR de 90 tornam-se NULL)

# Exclusão

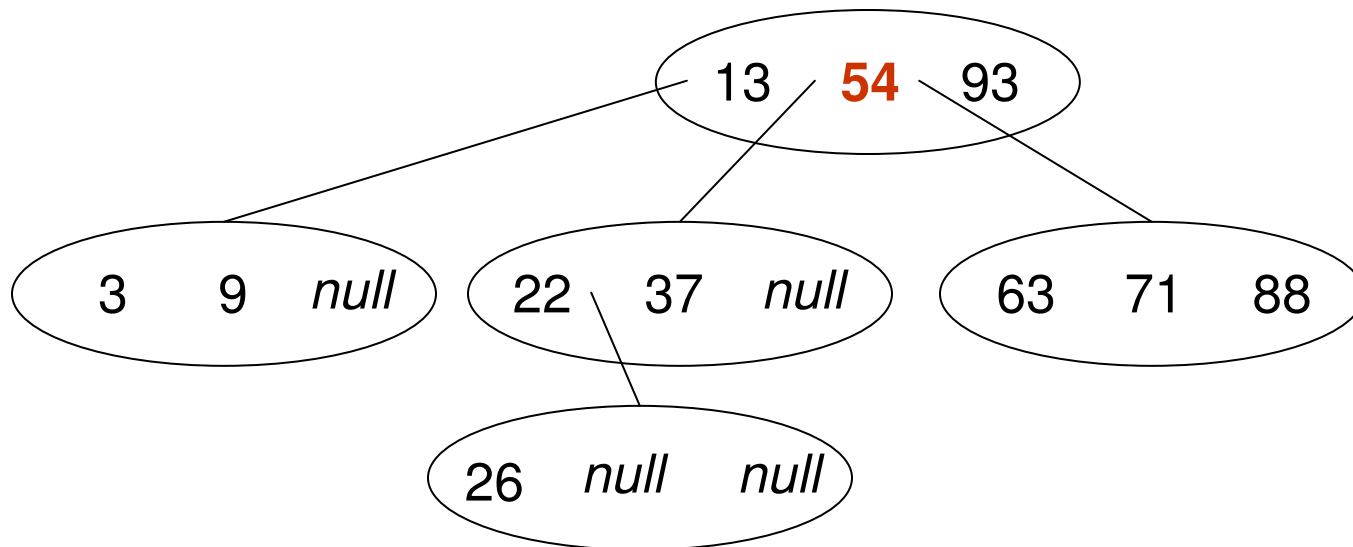
Excluir o elemento 90

→ não há subárvores ESQ e DIR e o elemento é o único do nodo



# Exclusão

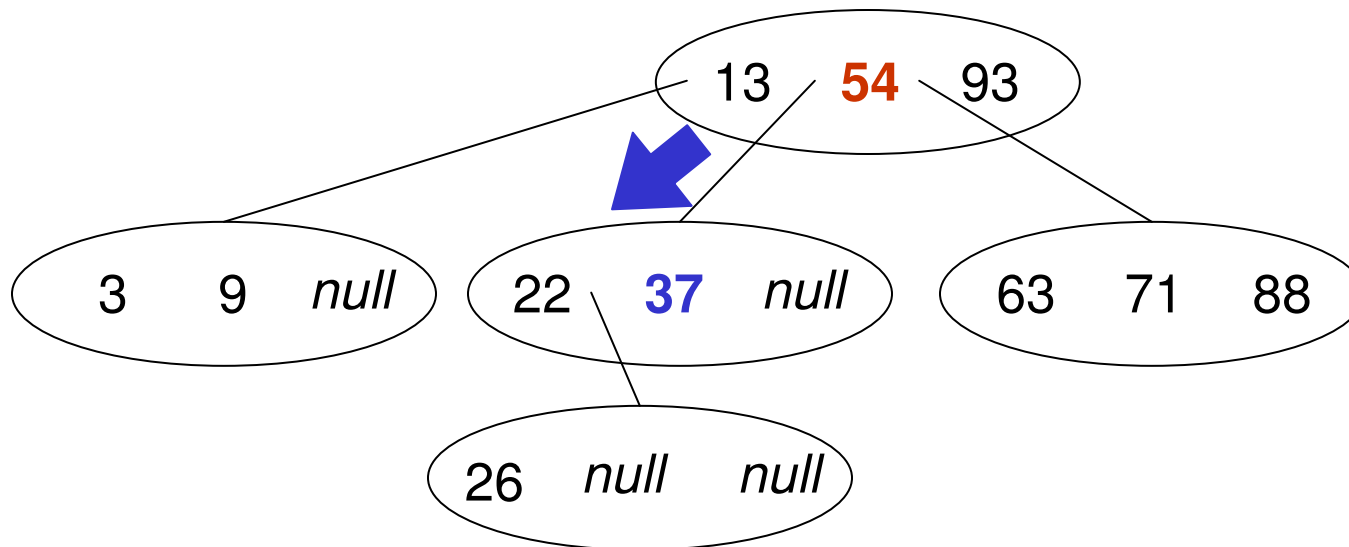
Excluir o elemento 54



# Exclusão

Excluir o elemento 54

→ existem subárvores ESQ e DIR: o elemento é trocado com o maior elemento na ESQ

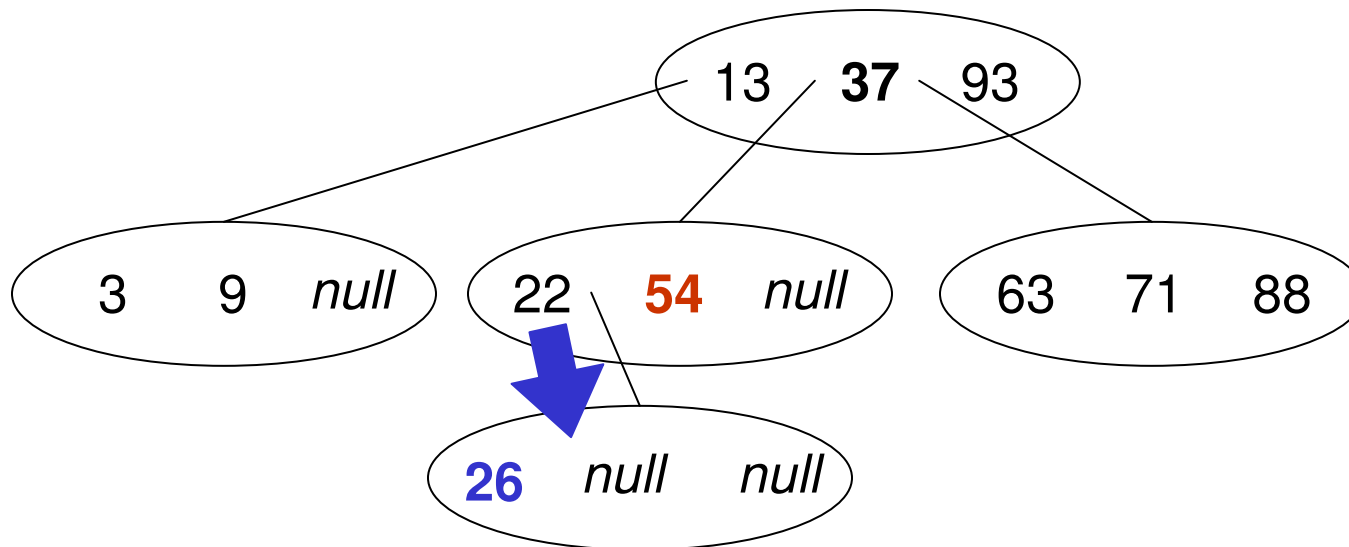




# Exclusão

Excluir o elemento 54

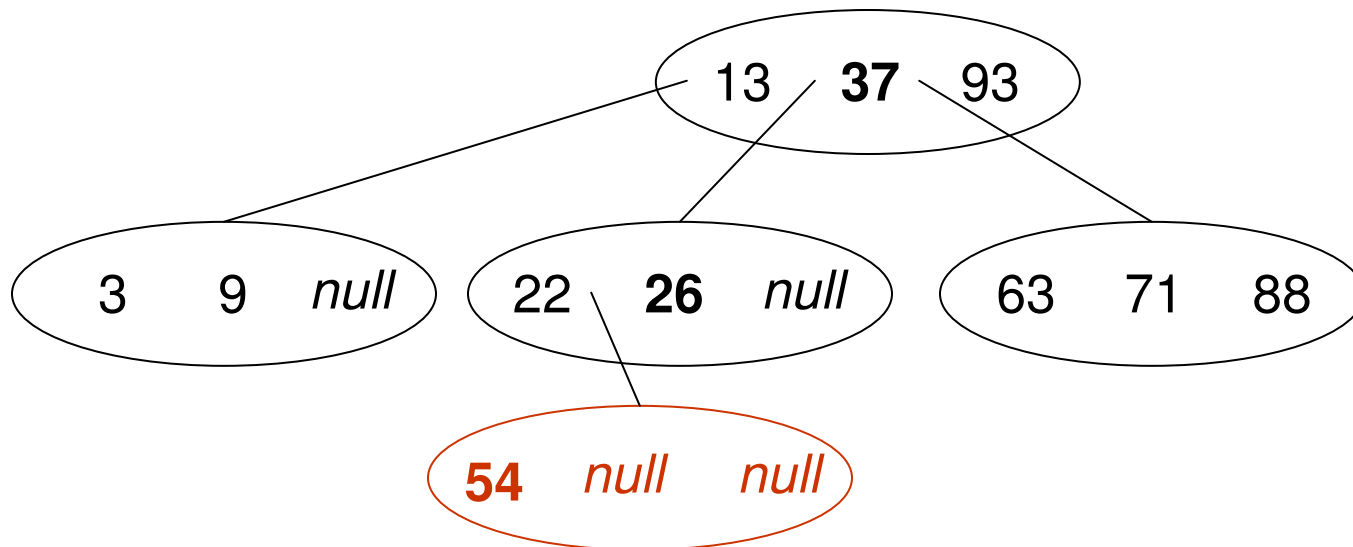
→ o elemento possui subárvore ESQ não vazia: nova troca.



# Exclusão

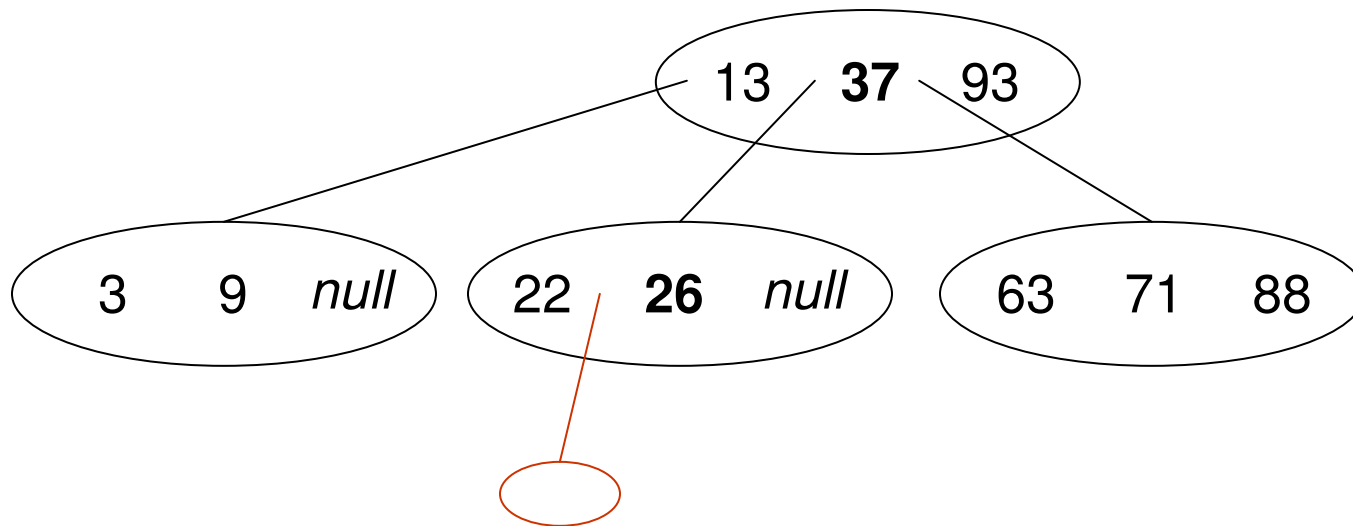
Excluir o elemento 54

→ o elemento pode ser removido



# Exclusão

Excluir o elemento 54





# Problema da Árvore N-ária de Pesquisa

- Uma Árvore N-ária de Pesquisa pode ficar **desbalanceada**.
- Exemplo:
  - $N = 4$
  - ordem de inclusão: 20-60-90-12-7-18-5-4-6-1-3

Alternativa de solução: **Árvore B**