

Algoritmos e Estruturas de Dados III

4.3 Árvores B+ e B*

2023



PUC Minas

Árvore B+

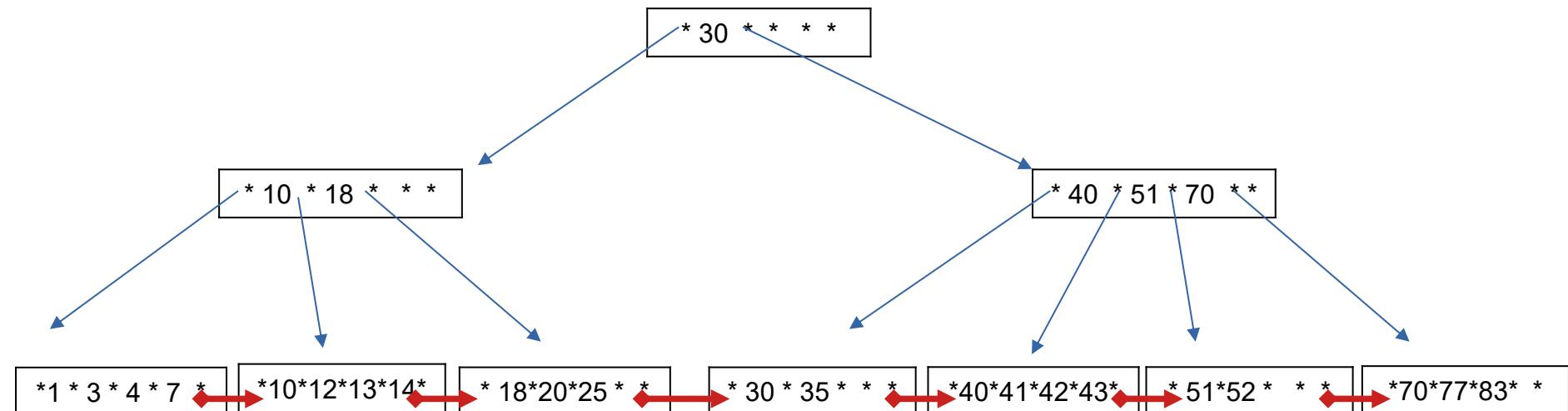


PUC Minas

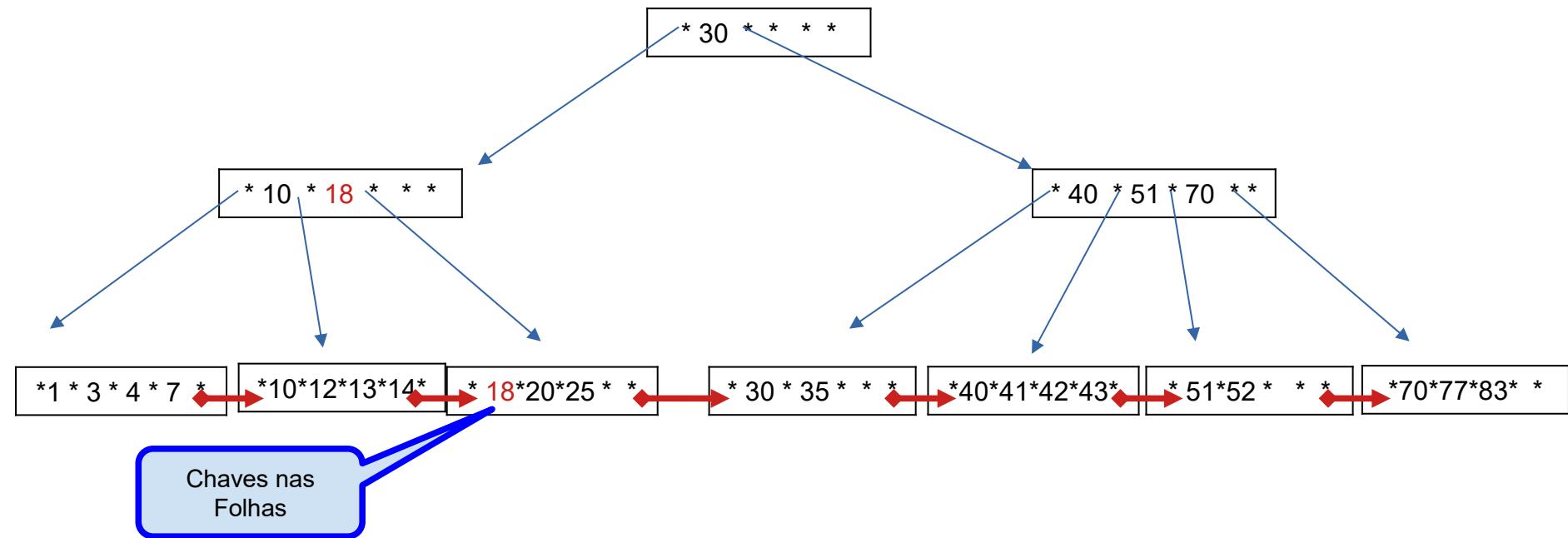
Árvore B+

- Todas as chaves são armazenadas nas folhas
- Cada folha aponta para a próxima folha (para permitir a leitura sequencial)
- As folhas podem possuir uma estrutura diferente das páginas não folhas, por serem as únicas páginas a carregarem dados
- Vantagens
 - Mantém a eficiência da busca e da inserção da árvore B
 - Aumenta a eficiência da localização do próximo registro na árvore de $O(\log n)$ para $O(1)$
 - Não é necessário manter nenhum ponteiro de registro em nós não-folha

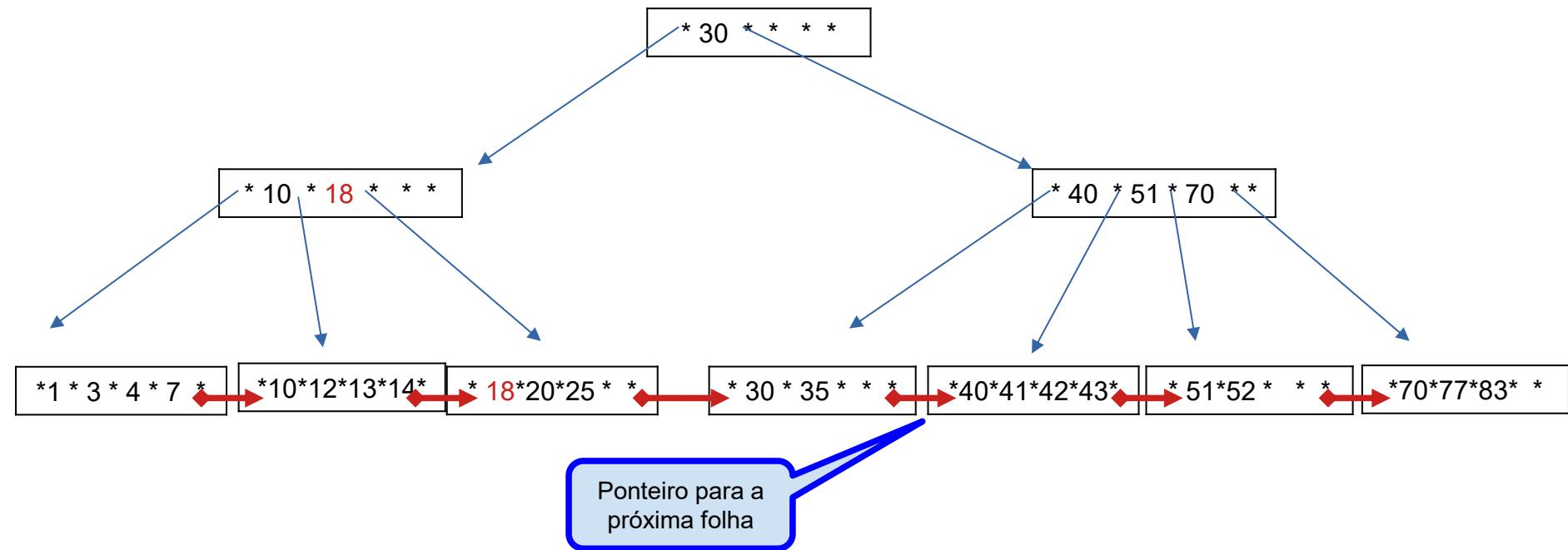
Árvore B+



Árvore B+



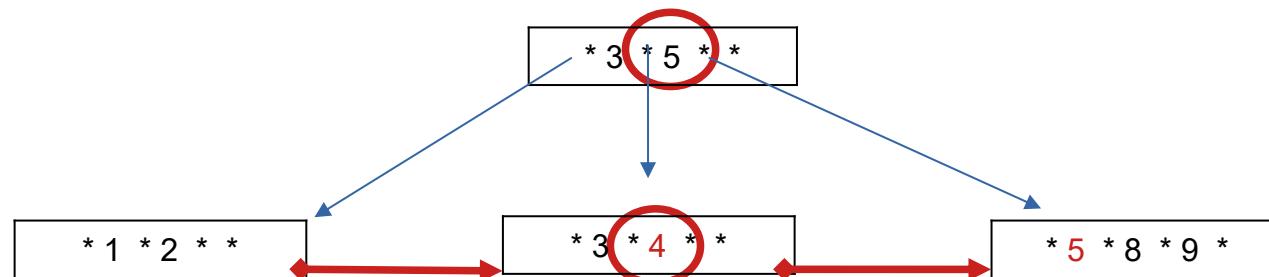
Árvore B+



Árvore B+ Operações

- As operações de busca, inserção e remoção são efetuadas de modo similar à Árvore B
- Uma busca por conjunto de chaves é simplificada à encontrar a primeira chave de interesse, seguida por uma operação de busca linear na lista ligada de folhas

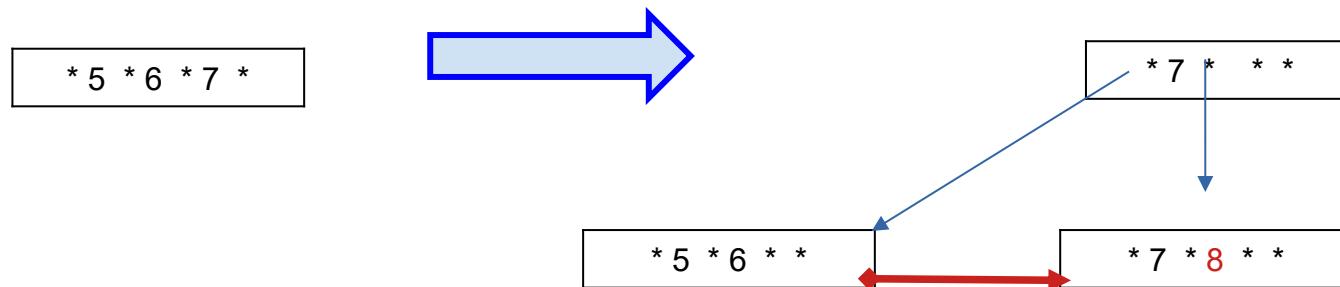
Exemplo: Retorne todos os registros entre [4, 5]



Árvore B+ Inserção

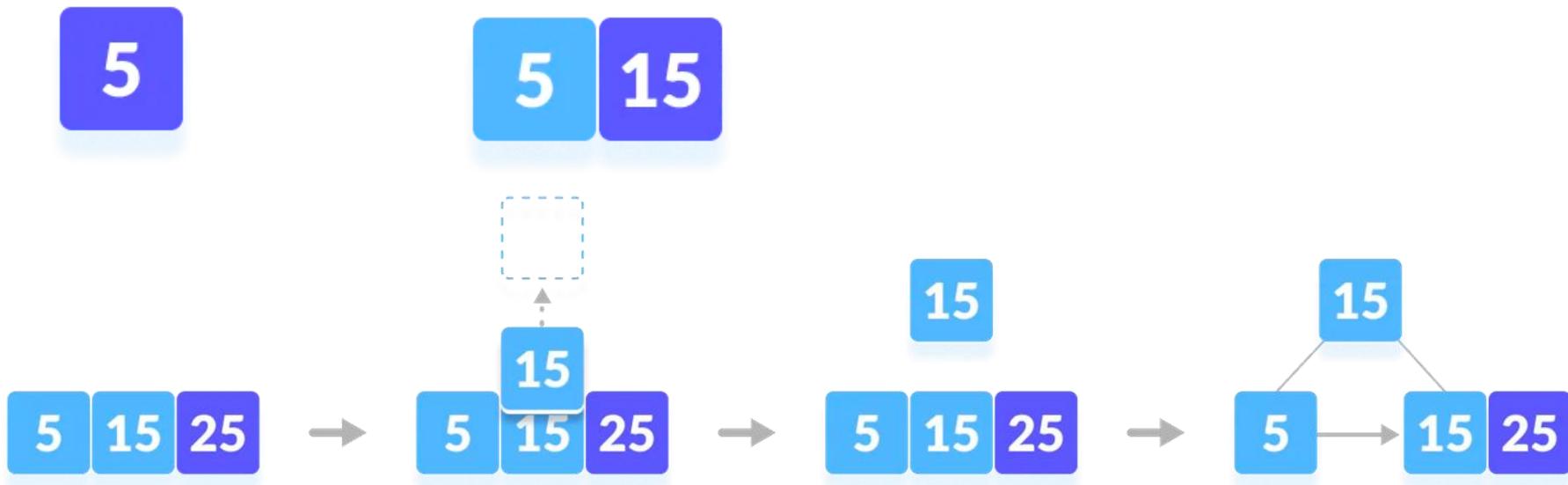
- Todas as chaves devem existir tanto em nó interno quanto em nó folha
- Ao ocorrer uma divisão de nó a chave mediana deve ser copiada para o novo nó pai e mantida no novo nó folha

Exemplo: Insira 8



Árvore B+ - Inserção

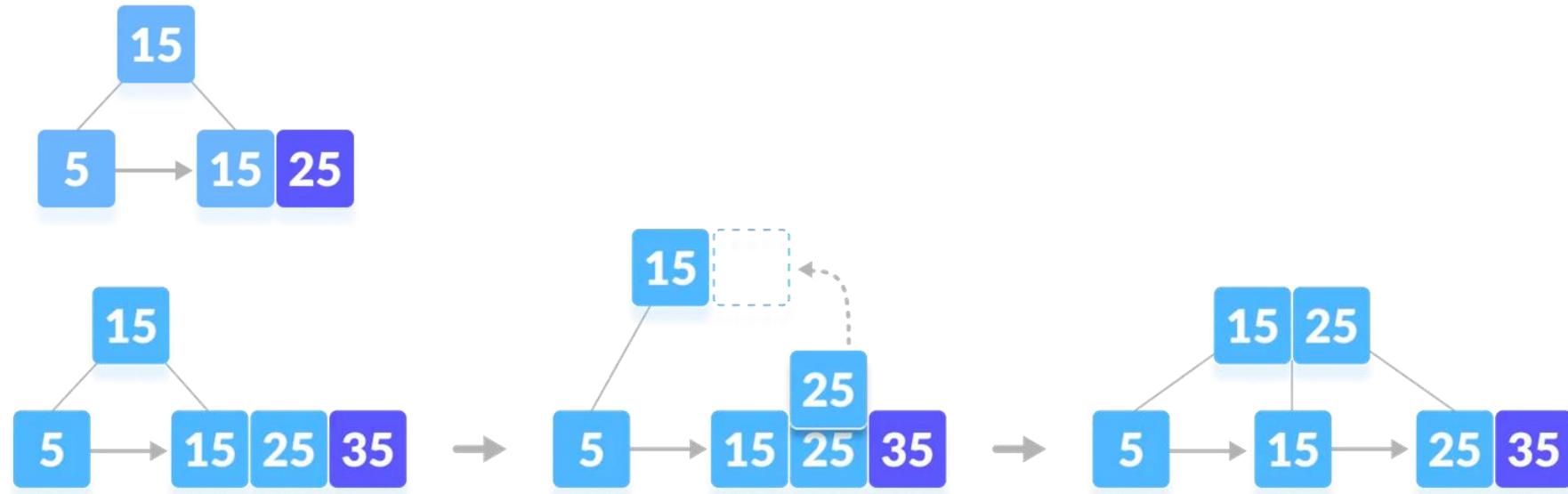
Ordem 3, inserção de: **5, 15, 25, 35, 45**



Fonte: <https://www.programiz.com/dsa/insertion-on-a-b-plus-tree>

Árvore B+ - Inserção

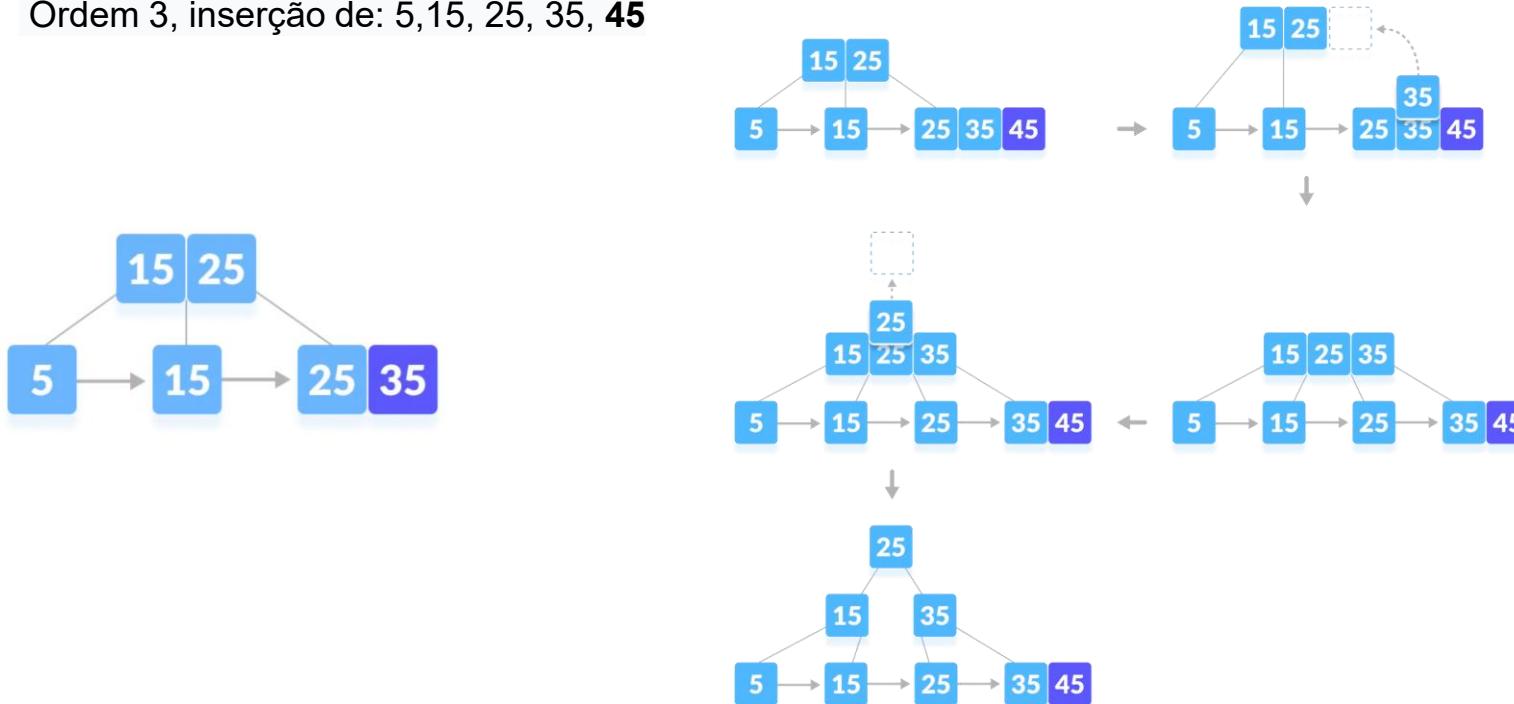
Ordem 3, inserção de: 5, 15, 25, **35**, 45



Fonte: <https://www.programiz.com/dsa/insertion-on-a-b-plus-tree>

Árvore B+ - Inserção

Ordem 3, inserção de: 5, 15, 25, 35, **45**

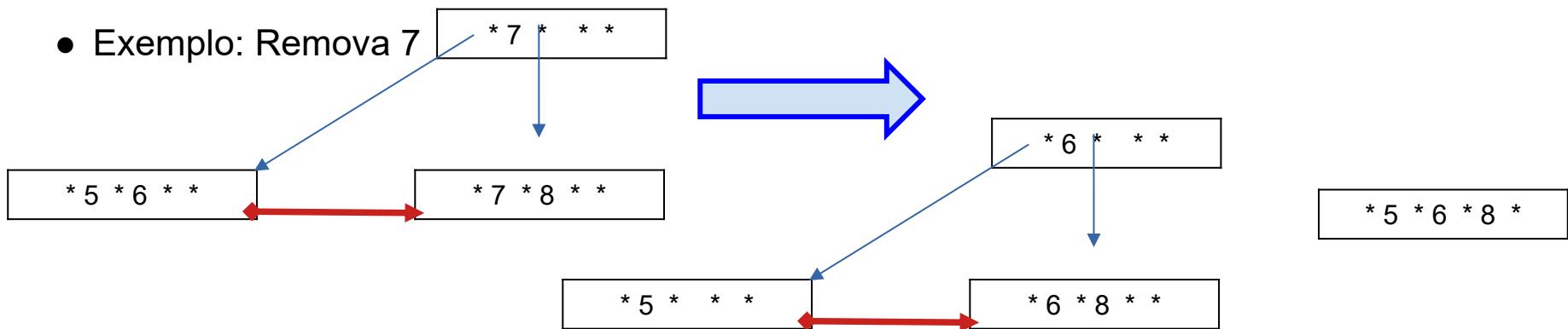


Fonte: <https://www.programiz.com/dsa/insertion-on-a-b-plus-tree>

Árvore B+ Remoção

- Se chave a ser removida estiver em um nó folha: basta executar a remoção normalmente
- Se chave a ser removida estiver em um nó interno: ela deve ser substituída por seu antecessor ou sucessor (depende do algoritmo). Buscá-lo em seu filho esquerdo ou direito (depende do algoritmo), fazer a substituição e então chamar a função de remoção recursivamente para a subárvore à direita.

- Exemplo: Remova 7



0	96											
8	2	-1	10	E5	-1	15	E16	-1				
96	1	184	9	E0	8							
184	2	-1	1	E3	-1	2	E4	-1				

Árvore B \times Árvore B+

Endereço no arquivo de dados

Ponteiro para a página posterior

0	96											
8	2	-1	9	E0	-1	10	E5	-1	15	E16	-1	
96	1	184	9	8								
184	2	-1	1	E3	-1	2	E4	8				

0

96

8

2

-1

10

E5

-1

15

E16

-1

96

1

184

9

E0

8

184

2

-1

1

E3

-1

2

E4

-1

Árvore B

x

Árvore B+ (duplamente encadeada)Ponteiro para a
página posteriorPonteiro para a
página anterior

0

96

8

2

184

9

E0

-1

10

E5

-1

15

E16

-1

96

1

184

9

8

184

2

-1

1

E3

-1

2

E4

8

Árvore B*



PUC Minas

Árvore B*

- Variação da árvore B proposta em 1973 por Knuth
- Técnica de redistribuição de chaves também é empregada durante as operações de inserção.
- Operação de split pode ser adiada até que duas páginas irmãs estejam completamente cheias

Para lembrar, na Operação de Inserção na Árvore B:

- Se o elemento couber na página, basta incluí-lo de forma ordenada
- Se não couber, a página deve ser dividida em duas (split) e o elemento do meio deve ser promovido

Árvore B*

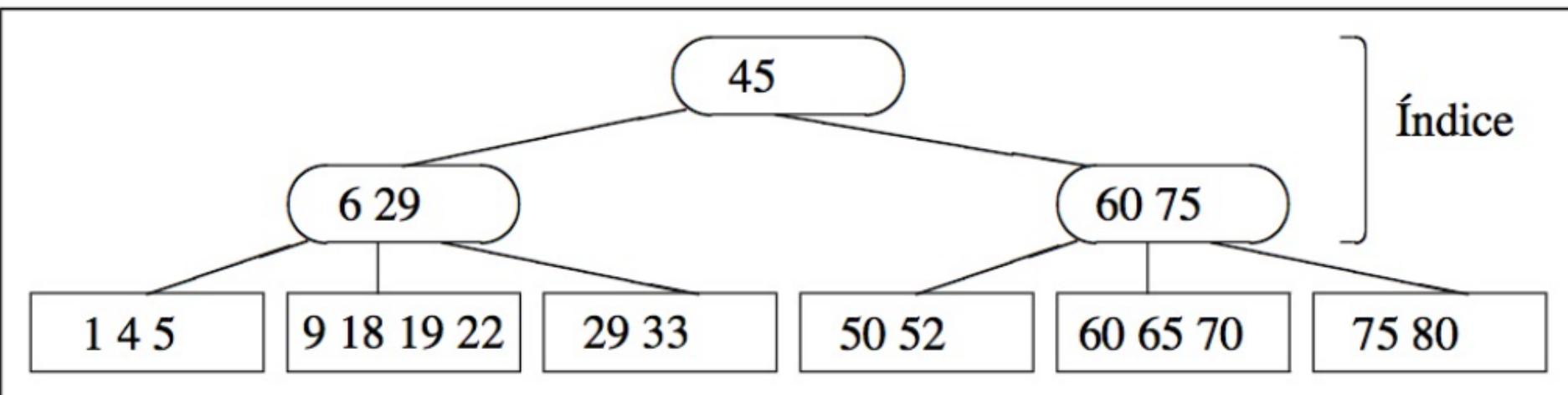


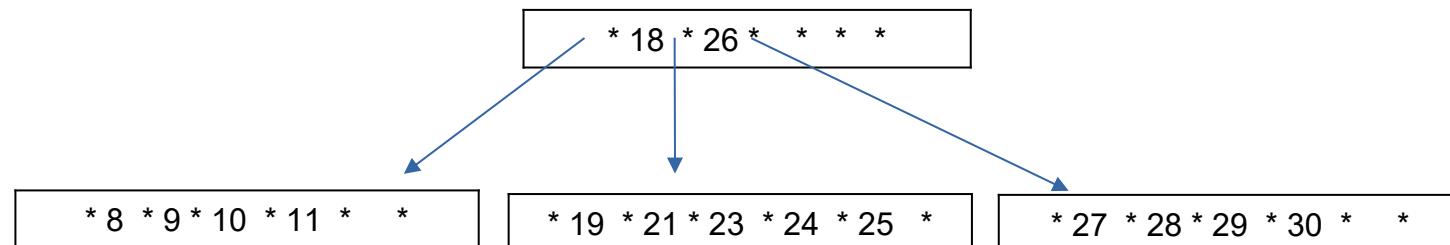
Figura 6.15 Exemplo de uma árvore B*.

Árvore B*

```
package cap6;
import cap4.Item; // vide Programa 4.1
public class ArvoreBEstrela {
    private static abstract class Pagina {
        int n; Item chaves[];
    }
    private static class PaginaInt extends Pagina {
        Pagina p[];
        public PaginaInt (int mm) {
            this.n = 0; this.chaves = new Item[mm];
            this.p = new Pagina[mm+1];
        }
    }
    private static class PaginaExt extends Pagina {
        Object registros[];
        public PaginaExt (int mm2) {
            this.n = 0; this.chaves = new Item[mm2];
            this.registros = new Object[mm2];
        }
    }
    private Pagina raiz;
    private int mm, mm2;
```

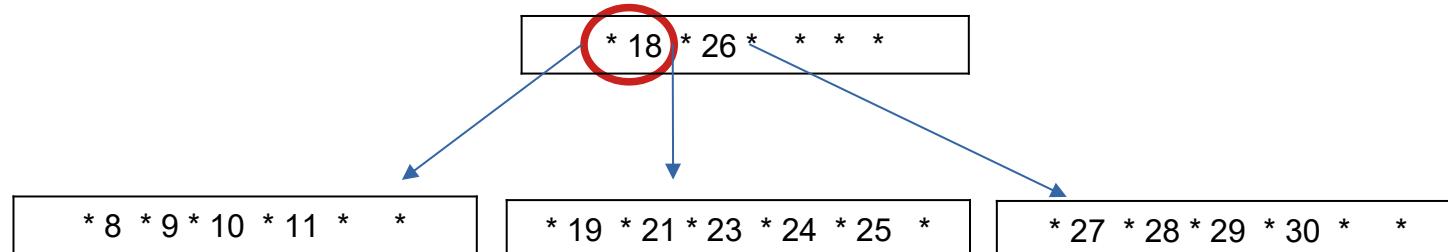
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



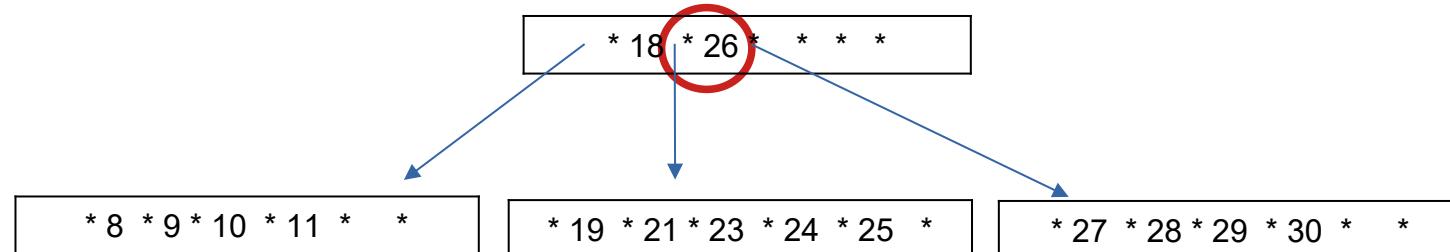
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



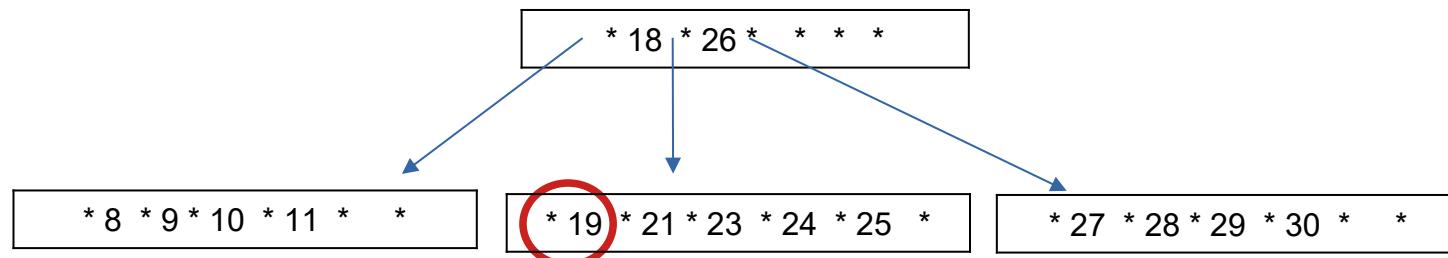
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



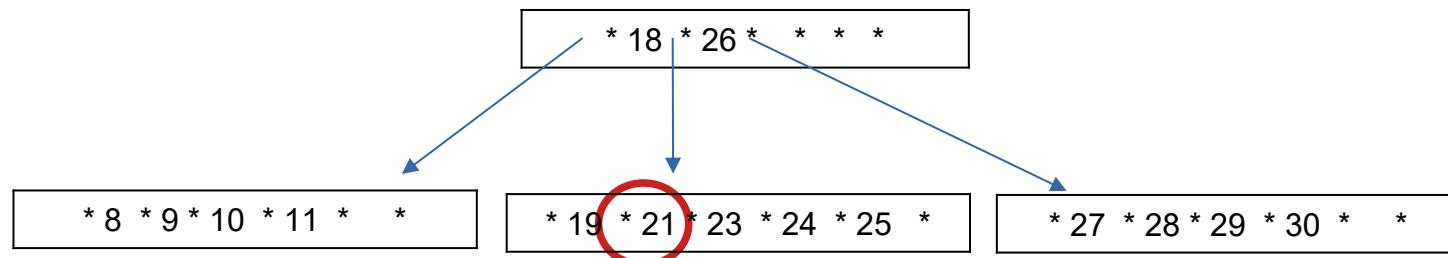
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



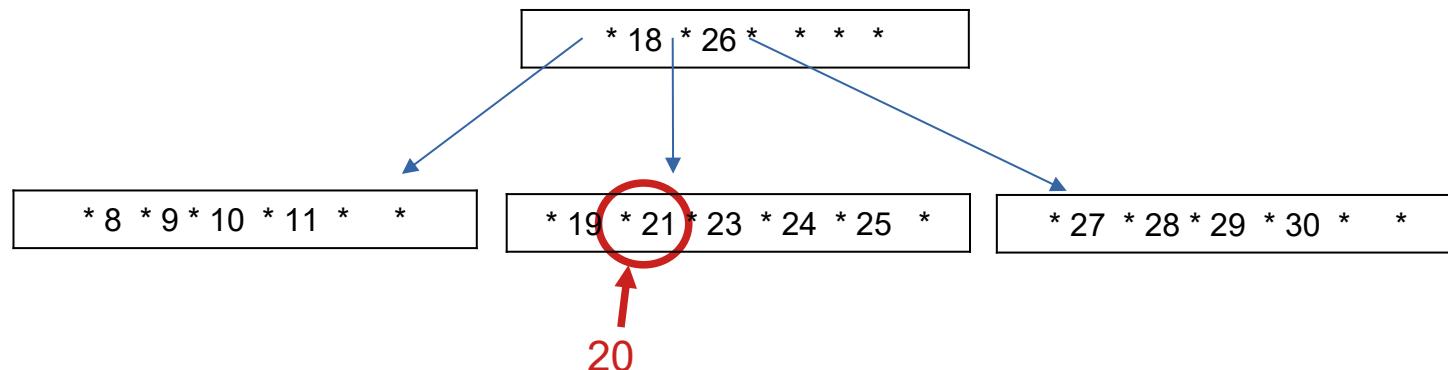
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



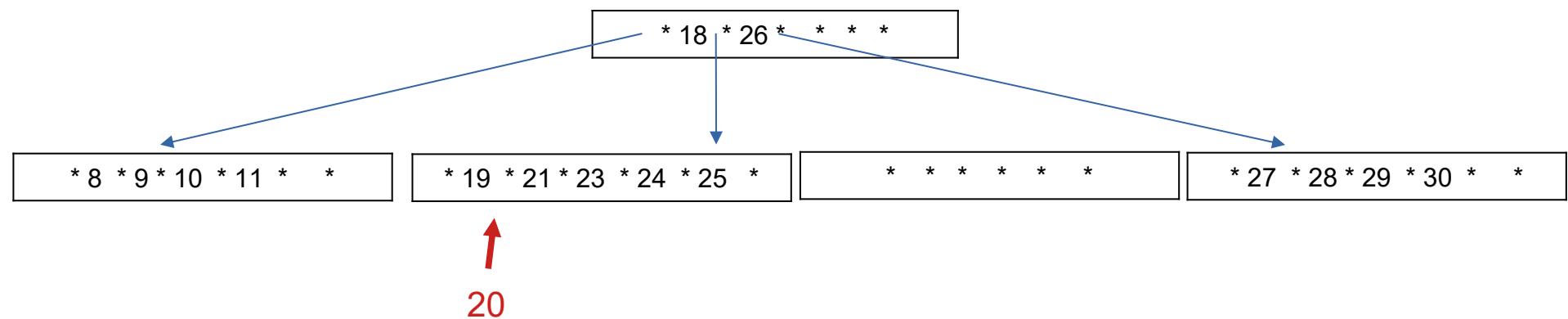
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



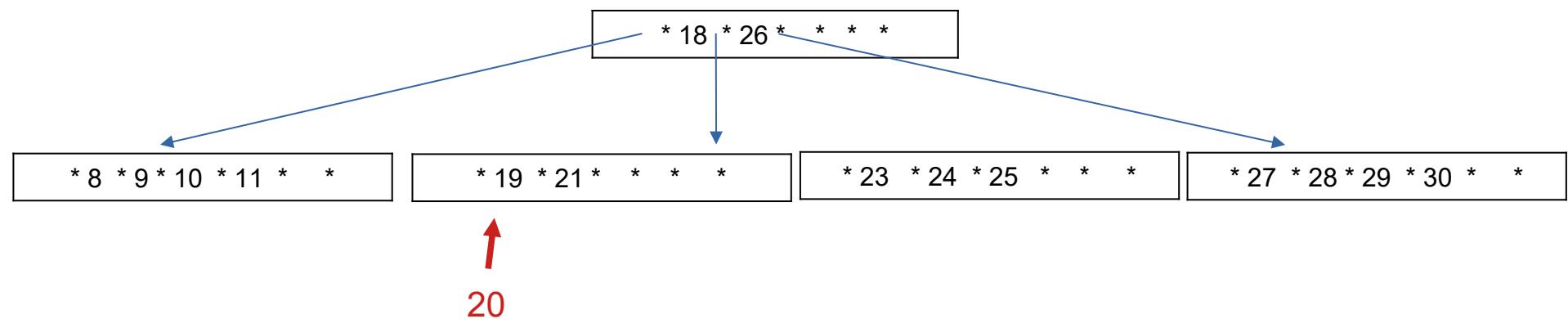
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



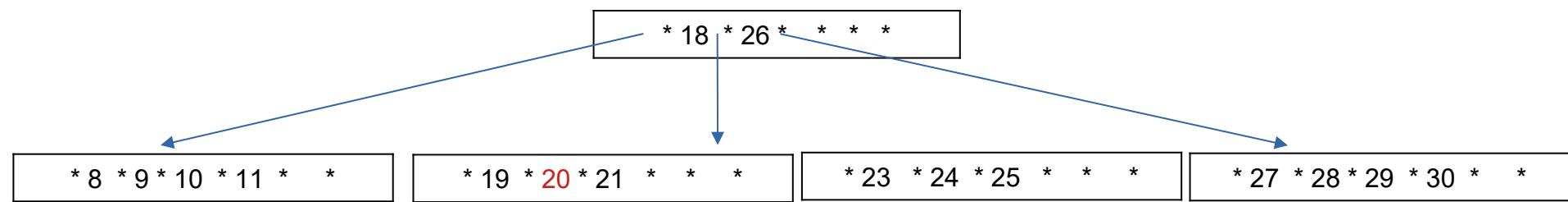
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



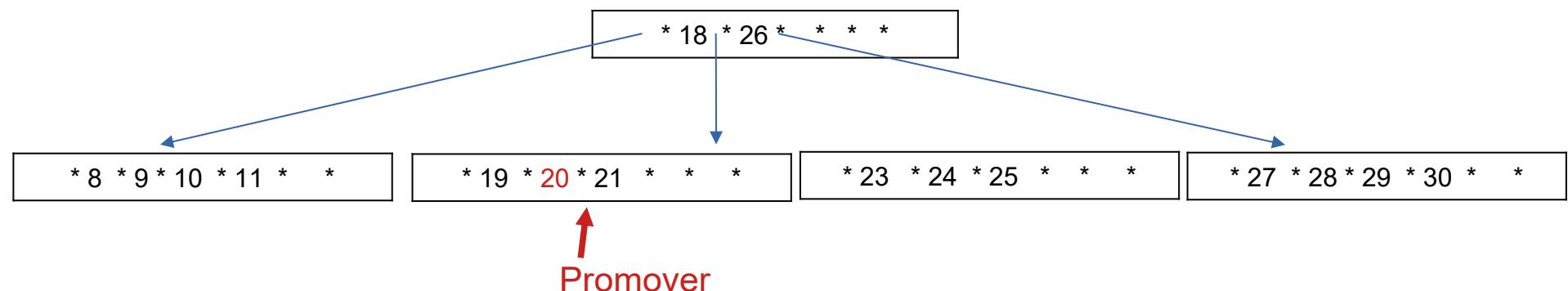
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



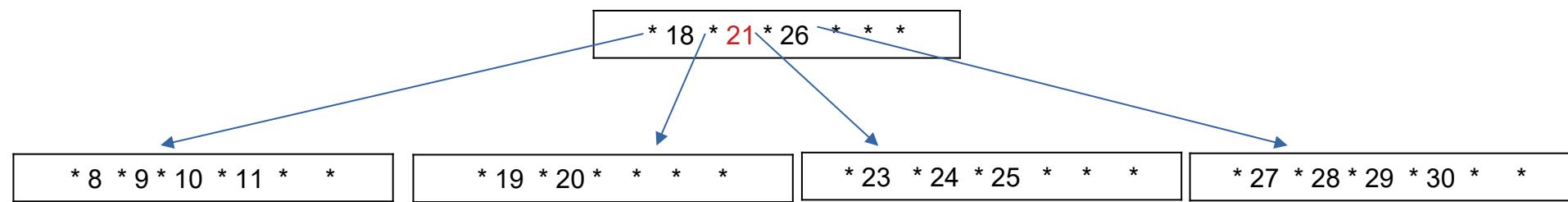
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



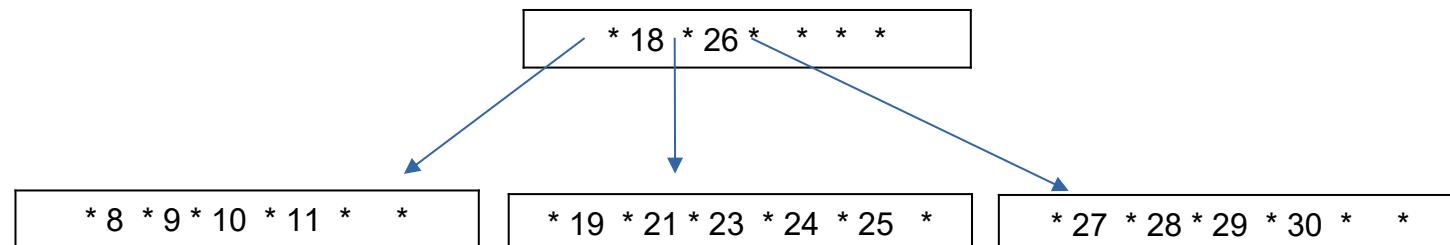
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B:
Divisão de Um pra Dois (**one-to-two split**)



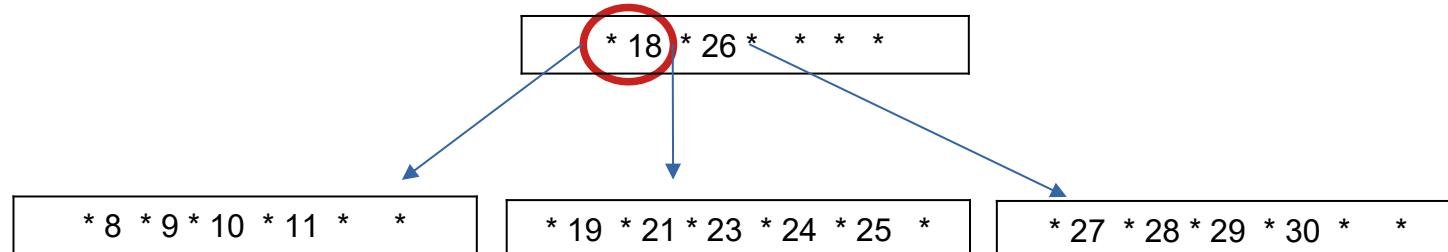
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



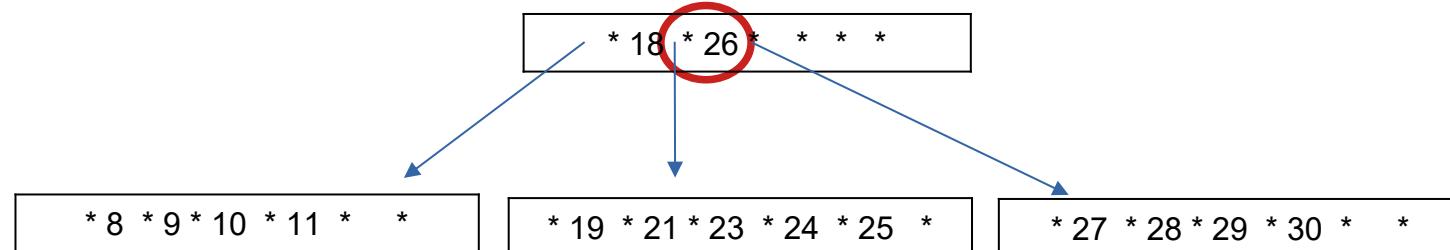
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



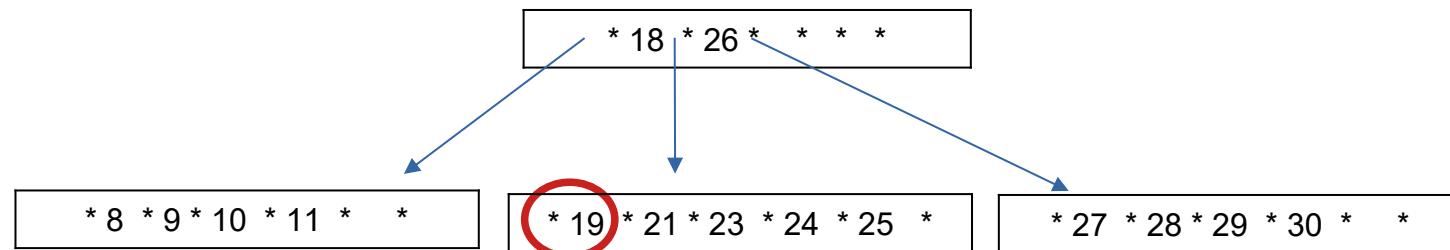
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



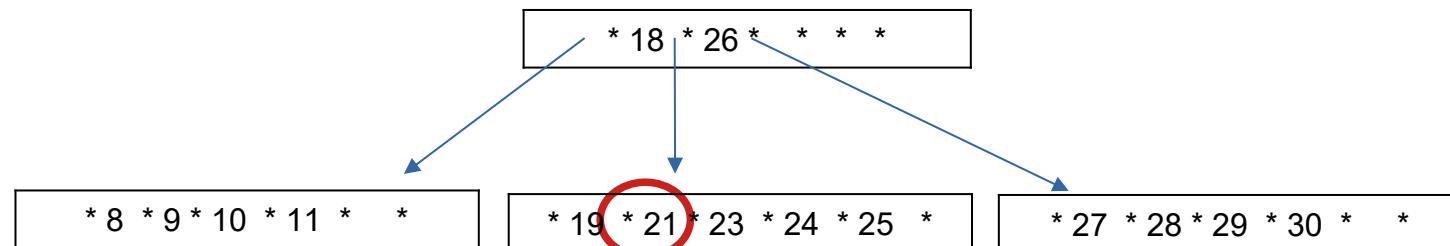
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



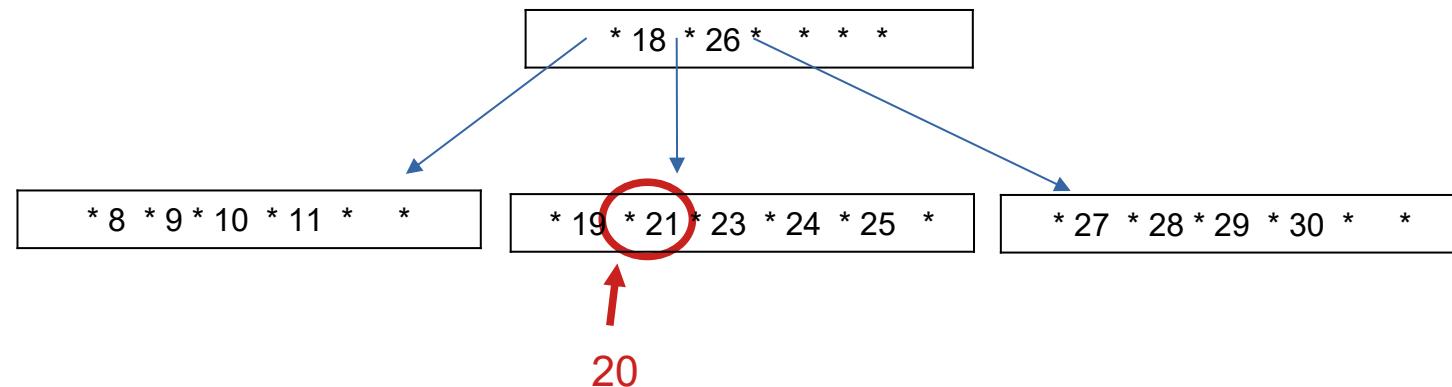
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



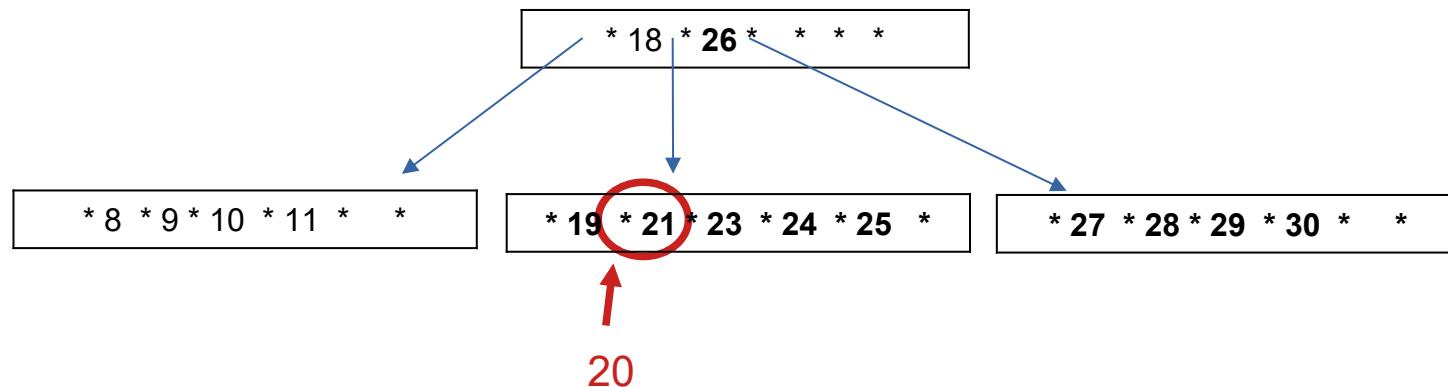
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



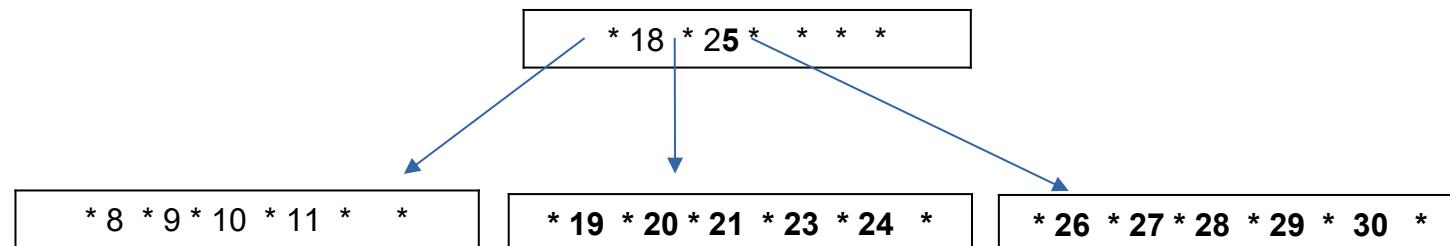
Árvore B* – Inclusão com Split

Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



Árvore B* – Inclusão com Split

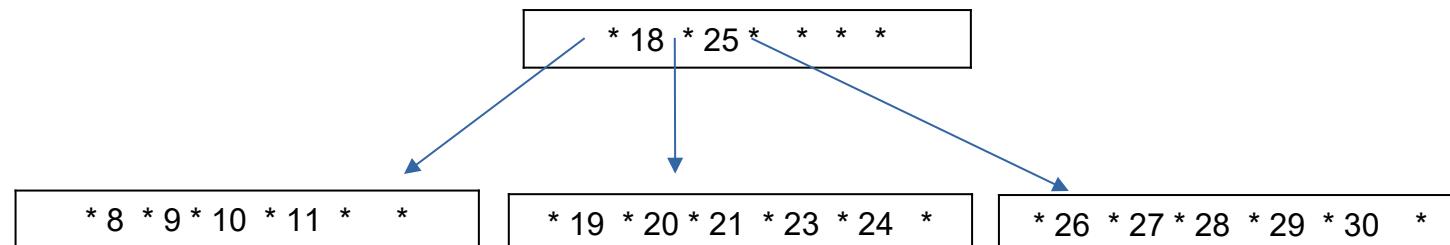
Inclusão da chave 20 em uma Árvore B*:
Redistribuição de Chaves entre irmãos



Árvore B* – Inclusão com Split

Inclusão da chave **22** em uma Árvore B*:

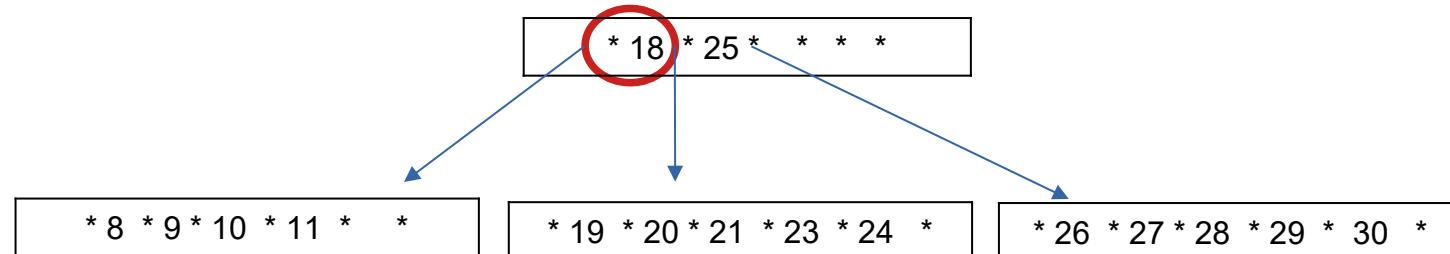
Divisão de Dois pra Três (two-to-three split)



Árvore B* – Inclusão com Split

Inclusão da chave 22 em uma Árvore B*:

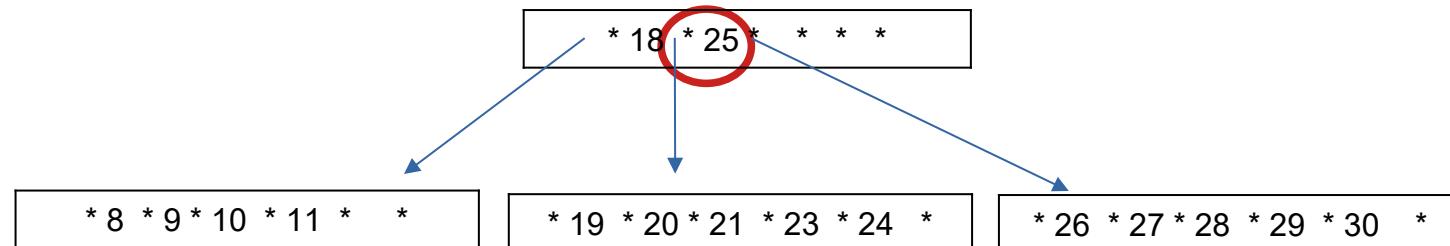
Divisão de Dois pra Três (two-to-three split)



Árvore B* – Inclusão com Split

Inclusão da chave 22 em uma Árvore B*:

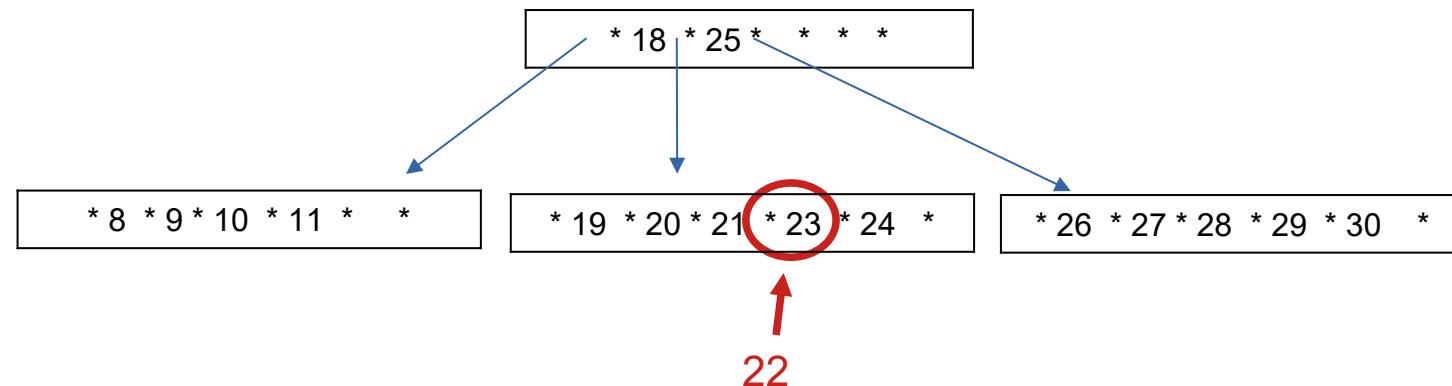
Divisão de Dois pra Três (two-to-three split)



Árvore B* – Inclusão com Split

Inclusão da chave 22 em uma Árvore B*:

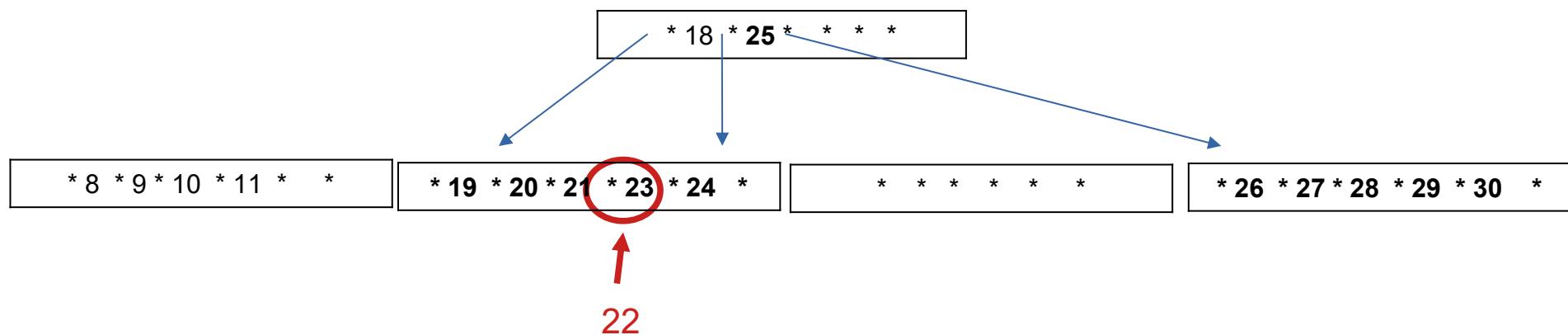
Divisão de Dois pra Três (two-to-three split)



Árvore B* – Inclusão com Split

Inclusão da chave 22 em uma Árvore B*:

Divisão de Dois pra Três (two-to-three split)

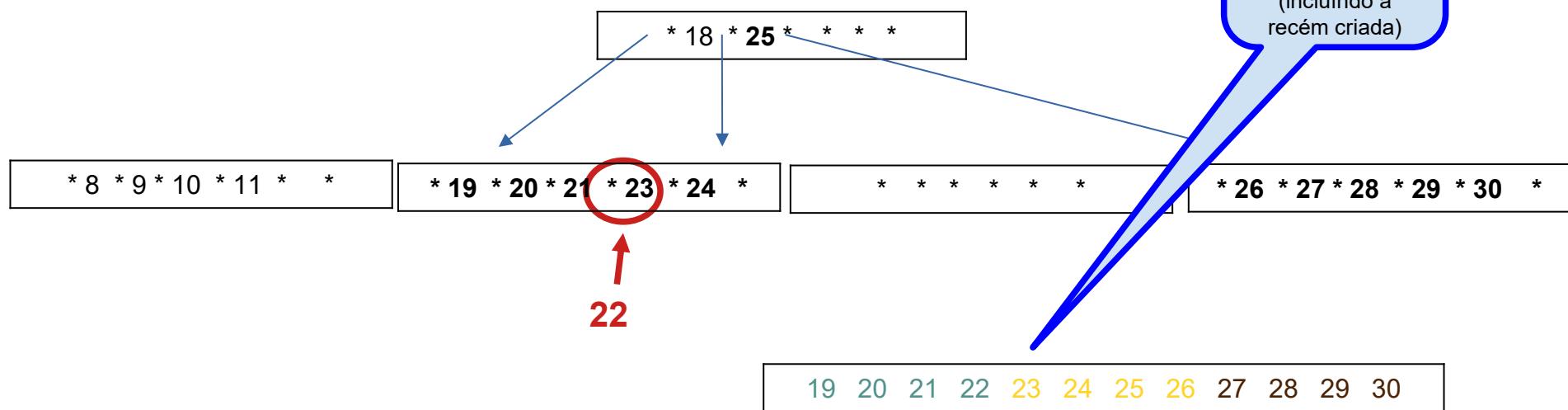


Árvore B* – Inclusão com Split

Inclusão da chave 22 em uma Árvore B*:

Divisão de Dois pra Três (two-to-three split)

Temos 12 chaves, incluindo a 22 para distribuir entre 3 páginas (incluindo a recém criada)

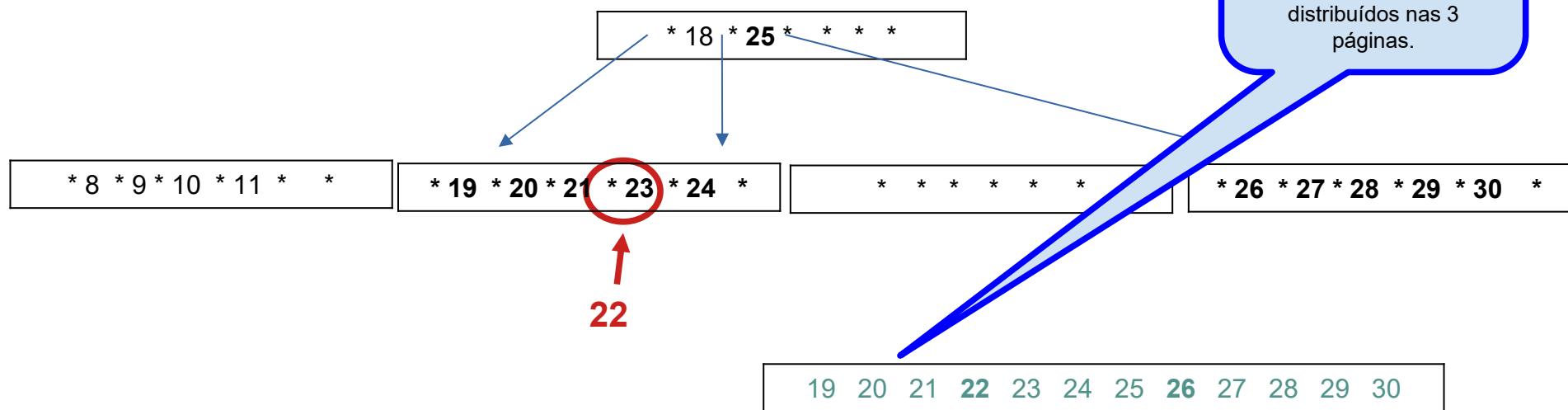


Árvore B* – Inclusão com Split

Inclusão da chave 22 em uma Árvore B*:

Divisão de Dois pra Três (two-to-three split)

Dividindo em 3 blocos de 4 elementos, identifico quais seriam os que devem ser promovidos e quais podem ser distribuídos nas 3 páginas.



Árvore B* – Inclusão com Split

Inclusão da chave 22 em uma Árvore B*:

Divisão de Dois pra Três (two-to-three split)

Promovidos



* 19 * 20 * 21 * * *

* 23 * 24 * 25 * * *

* 27 * 28 * 29 * 30 * *

Árvore B* - Propriedades

- Em uma Árvore B* de ordem m
- Cada página tem no máximo m descendentes
- Toda página, exceto a raiz e as folhas, tem no mínimo $(2m-1)/3$ descendentes
- A raiz tem pelo menos 2 descendentes, a menos que seja uma folha
- Todas as folhas estão no mesmo nível
- Uma página não-folha com k descendentes contém $k-1$ chaves
- Uma página folha contém no mínimo piso($(2m-1)/3$) e no máximo $m-1$ chaves

Árvore B* - Propriedades

- As principais alterações estão na segunda e na última regra
- Esta propriedade afeta as regras para remoção e redistribuição
- Deve-se tomar cuidado na implementação, uma vez que a raiz nunca tem irmã, e portanto requer tratamento especial
- Uma solução é dividir a raiz usando a divisão convencional (one-to-two split), outra é permitir que a raiz seja maior