

# B-Árvores

Siang Wun Song - Universidade de São Paulo - IME/USP

MAC 5710 - Estruturas de Dados - 2008

Os slides sobre este assunto são parcialmente baseados nos artigos

- Bayer, R. and E. McCreight. Organization and maintenance of large ordered indexes. *Acta Informatica*, Vol. 1, 1972, pp. 173-189.
- D. Comer. The ubiquitous B-tree. *ACM Computing Surveys*, June 1979, pp. 121-137.

# Busca de dados em memória secundária (disco)

- Árvores binárias de busca, balanceadas ou não, não são adequadas para o armazenamento e busca de dados em memória secundária (como disco rígido).
- O acesso a disco envolve um posicionamento da cabeça do disco, além da transferência de dados propriamente ditos. O posicionamento depende do tempo de rotação do disco que é da ordem de 8 mili-segundos. Um acesso a disco leva tipicamente 10 a 15 mili-segundos, o que é considerável em comparação com o tempo de acesso à memória primária (RAM), da ordem de 100 nano-segundos. No tempo para acessar uma vez o disco, podemos fazer cerca de 100.000 acessos à memória.
- Mesmo em árvores balanceadas, de  $n$  chaves,  $O(\log n)$  acessos a disco podem ser excessivos. Para uma árvore binária de busca balanceada de  $n = 1$  milhão de chaves armazenadas em disco,  $\log n = 20$  acessos a disco podem ser considerados custosos demais.

# Introdução a B-árvore

- Como um acesso a disco é uma operação cara, então ao invés de buscar um dado de cada vez, procura-se transferir, em cada acesso, uma quantidade maior de dados. Um nó da B-árvore (chamado página) contém tipicamente centenas de chaves.
- B-árvore pode ser considerada uma generalização de uma árvore binária de busca balanceada. De fato, ela é uma extensão da chamada árvore-2-3.
- B-árvore é uma estrutura de dado muito bem sucedida, com diversas variantes usadas na implementação de bases de dados de uso comercial, como Oracle, Sybase, VSAM da IBM, dBASE, etc.
- Os criadores da B-árvore (R. Bayer e E. McCreight) não explicam o por quê do nome B-árvore. Conjectura-se que a letra “B” lembra “B”ayer, ou “B”alanceada ou ainda “B”oeing, a companhia onde trabalhavam os dois autores.

# Introdução a B-árvore

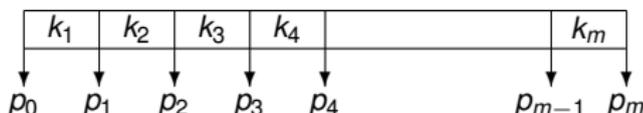
Um nó da B-árvore é também conhecido pelo nome de página. Veremos que cada página pode conter uma grande quantidade de chaves.

- A chave exerce um papel importante na busca, pois ela identifica unicamente um elemento de informação. Naturalmente, além da chave, podemos ter outras informações associadas. Por exemplo, a chave pode ser o número CPF e a cada CPF podemos juntar ainda dados pessoais e rendimentos obtidos num ano fiscal.
- Nos exemplos representaremos apenas a chave, mas subentende-se que pode haver demais informações associadas. Essas informações podem estar armazenadas junto com a chave, ou em outros locais, sendo acessíveis por ponteiros armazenados junto com a chave.

# B-árvore de ordem $b$

Uma B-árvore de ordem  $b$  possui as seguintes propriedades:

- 1 Cada página contém no máximo  $2b$  chaves.
- 2 Cada página, exceto a página raiz, contém no mínimo  $b$  chaves.
- 3 Uma página com  $m$  chaves  $k_1 < k_2 < \dots < k_m$  possui  $m + 1$  ponteiros  $p_0, p_1, \dots, p_m$ . Só há duas situações possíveis:
  - A página é uma folha e não tem filhos: **todos** os ponteiros  $p_i$ ,  $0 \leq i \leq m$  apontam para nil.
  - A página não é folha e possui  $m + 1$  filhos apontados por  $p_i$ ,  $0 \leq i \leq m$ . **Nenhum** ponteiro é nil.

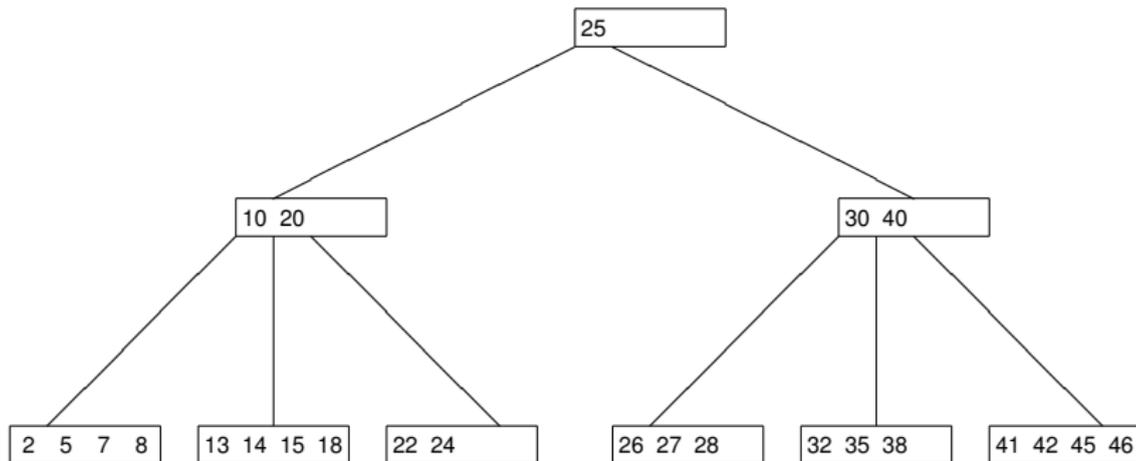


- Para toda chave  $k$  na subárvore apontada por  $p_0$ ,  $k < k_1$ .
- Para toda chave  $k$  na subárvore apontada por  $p_m$ ,  $k > k_m$ .
- Para toda chave  $k$  na subárvore apontada por  $p_i$ ,  $1 \leq i < m$ ,  $k_i < k < k_{i+1}$ .

- 4 Todas as páginas folhas aparecem no mesmo nível.

# Exemplo de uma B-árvore de ordem $b = 2$

A raiz de uma B-árvore de ordem  $b=2$  pode ter de 1 a  $2b = 4$  chaves;  
as demais páginas podem ter de  $b = 2$  a  $2b = 4$  chaves.



# Caso particular: árvore-2-3

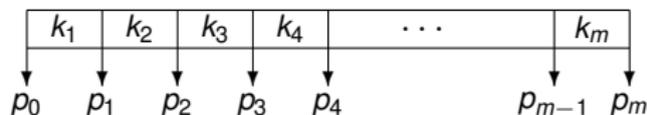
Um caso particular de B-árvore é a chamada árvore-2-3.

- Uma árvore-2-3 é uma B-árvore de ordem  $b = 1$ .
- Cada nó da árvore-2-3 tem 1 ou 2 chaves.
- Cada nó da árvore-2-3 tem 2 ou 3 filhos, daí o nome.
- A árvore-2-3 é uma árvore usada fazer busca de dados armazenados na memória principal.
- Para armazenamento e busca em disco, uma B-árvore usa uma ordem  $b$  grande, tipicamente de alguma centenas de chaves.

# Busca na B-árvore de ordem $b$

A busca de uma dada chave  $x$  numa B-árvore é análoga à busca na árvore binária de busca. A busca começa pela página raiz. É usual manter a raiz sempre na memória, evitando um acesso ao disco.

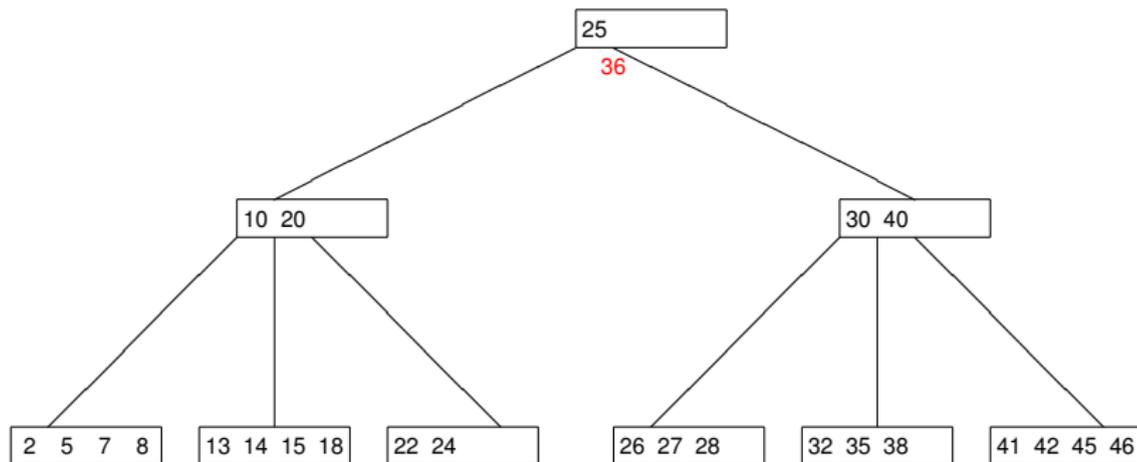
Estando em uma página da B-árvore, procedemos assim:



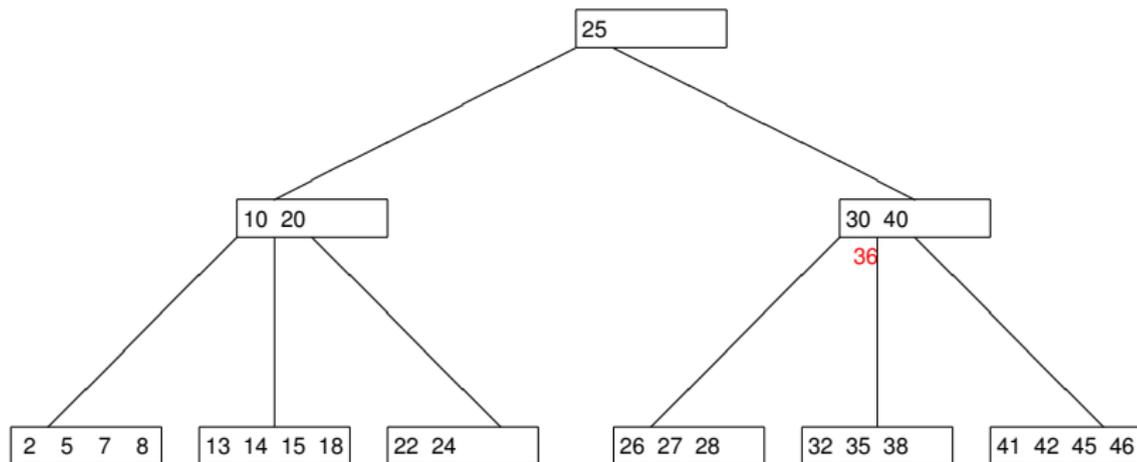
- Busca-se  $x$  na página corrente, usando um método de busca seqüencial ou busca binária, dependendo do valor de  $b$ . Para pequeno valores de  $b$ , busca seqüencial já basta.
- Se  $x$  estiver na página, então a busca termina.
- Se  $x < k_1$ , então continua a busca na página apontada por  $p_0$ .
- Se  $k_i < x < k_{i+1}$ , então continua a busca na página apontada por  $p_i$ .
- Se  $x > k_m$ , então continua a busca na página apontada por  $p_m$ .

Pode-se ver que a busca leva tempo  $O(\log_b n)$ , onde  $b$  é a ordem da B-árvore e  $n$  é o número total de chaves.

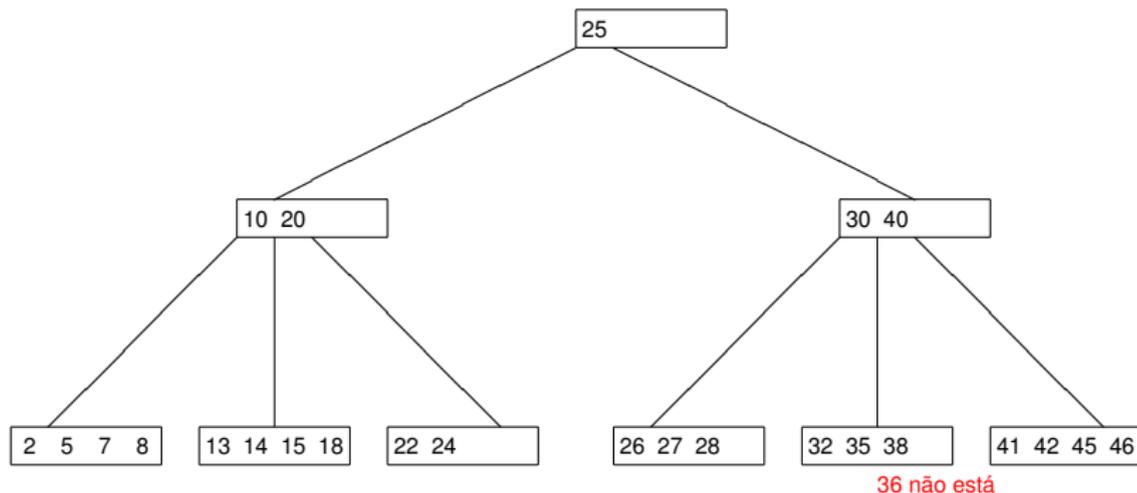
# Exemplo de busca na B-árvore: buscar chave 36



# Exemplo de busca na B-árvore: buscar chave 36



# Exemplo de busca na B-árvore: buscar chave 36



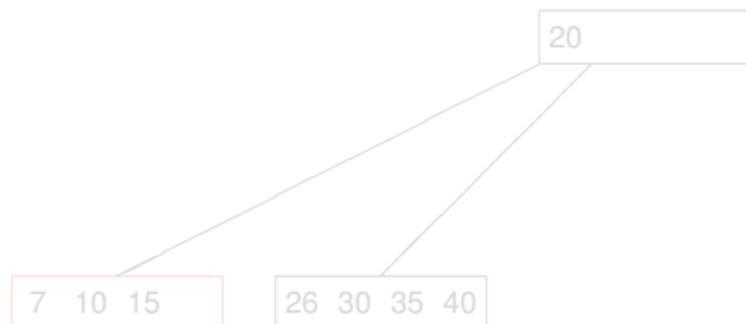
# Inserção na B-árvore de ordem $b$

Em uma B-árvore de ordem  $b$  cada nó (exceto a raiz) contém entre  $b$  a  $2b$  chaves. A raiz pode conter entre 1 a  $2b$  chaves.

Para inserir uma nova chave  $x$  numa B-árvore de ordem  $b$ :

- Primeiro localizamos a página folha onde será feita a inserção.

Exemplo: Para inserir 18, primeiro localizamos a página (em vermelho).



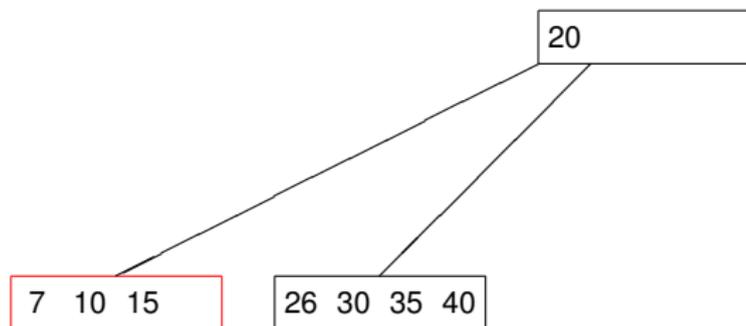
# Inserção na B-árvore de ordem $b$

Em uma B-árvore de ordem  $b$  cada nó (exceto a raiz) contém entre  $b$  a  $2b$  chaves. A raiz pode conter entre 1 a  $2b$  chaves.

Para inserir uma nova chave  $x$  numa B-árvore de ordem  $b$ :

- Primeiro localizamos a página folha onde será feita a inserção.

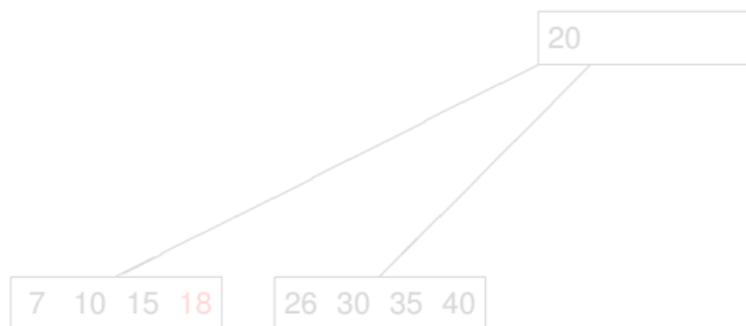
Exemplo: Para inserir 18, primeiro localizamos a página (em vermelho).



# Inserção na B-árvore de ordem $b$

- Verificamos quantas chaves já estão na página antes de adicionar a chave  $x$  na mesma.
- Caso 1: A página contém  $< 2b$  chaves, então basta inserir a nova chave  $x$  na página.

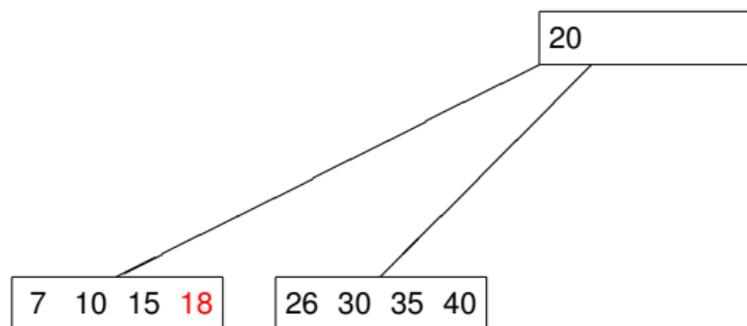
Exemplo: inserir 18 - A página contém  $3 < 2b = 4$  chaves, então basta inserir.



# Inserção na B-árvore de ordem $b$

- Verificamos quantas chaves já estão na página antes de adicionar a chave  $x$  na mesma.
- Caso 1: A página contém  $< 2b$  chaves, então basta inserir a nova chave  $x$  na página.

Exemplo: inserir 18 - A página contém  $3 < 2b = 4$  chaves, então basta inserir.



# Inserção na B-árvore de ordem $b$

- Caso 2: Antes de inserir a nova chave, a página já contém  $2b$  chaves. Adicionando mais a chave  $x$  iria resultar numa página com  $2b + 1$  chaves.
  - Consideramos as  $2b + 1$  chaves (incluindo a nova chave  $x$ ) em ordem crescente. Inserimos a chave do meio (mediana) na página pai, alocamos as primeiras  $b$  chaves numa página e as últimas  $b$  chaves noutra. Diz-se que há uma divisão ou cisão da página (em duas).

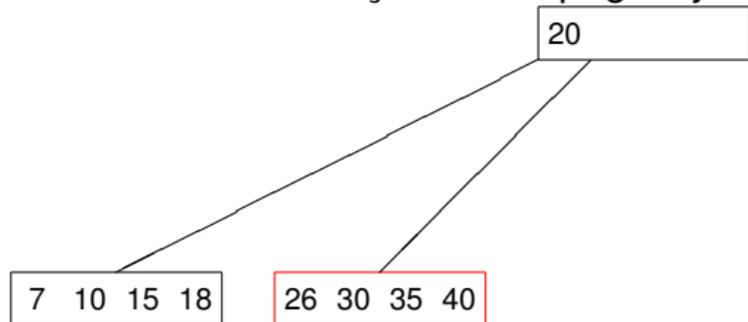
Exemplo: Para inserir 22, primeiro localizamos a página folha onde será feita a inserção. Essa página já tem  $2b = 4$  chaves.



# Inserção na B-árvore de ordem $b$

- Caso 2: Antes de inserir a nova chave, a página já contém  $2b$  chaves. Adicionando mais a chave  $x$  iria resultar numa página com  $2b + 1$  chaves.
  - Consideramos as  $2b + 1$  chaves (incluindo a nova chave  $x$ ) em ordem crescente. Inserimos a chave do meio (mediana) na página pai, alocamos as primeiras  $b$  chaves numa página e as últimas  $b$  chaves noutra. Diz-se que há uma divisão ou cisão da página (em duas).

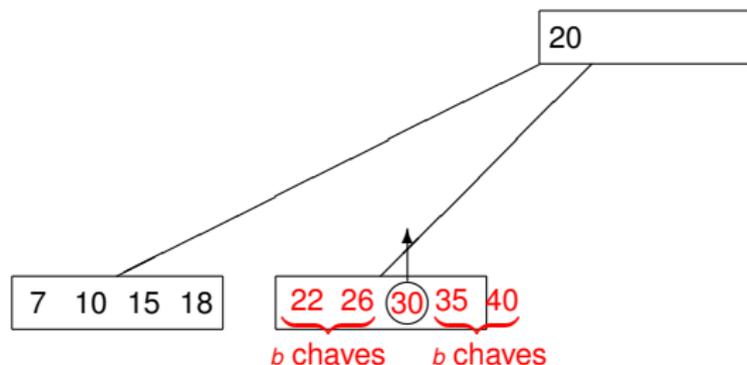
Exemplo: Para inserir 22, primeiro localizamos a página folha onde será feita a inserção. Essa página já tem  $2b = 4$  chaves.



# Inserção na B-árvore de ordem $b$

Com a inserção de uma nova chave, a página tem  $2b + 1 = 5$  chaves. Alocamos a chave do meio para pai,  $b$  chaves em uma página e  $b$  chaves na outra.

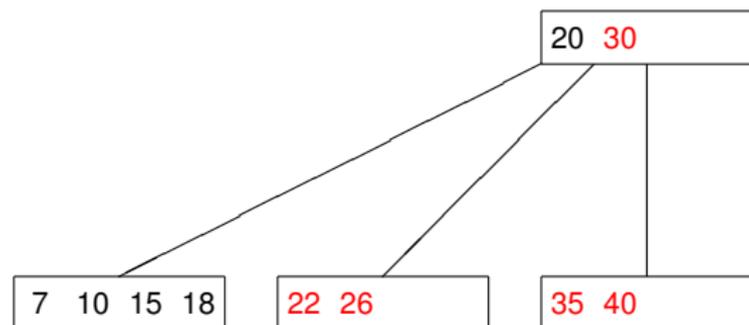
Exemplo: Movemos 30 para cima e separamos as duas metades em duas páginas (divisão ou cisão).



# Inserção na B-árvore de ordem $b$

Exemplo: inserir 22

A B-árvore após a divisão ou cisão da página.



# Inserção na B-árvore de ordem $b$

- No caso 2, a inserção de uma chave na página pai pode por sua vez, de forma recursiva, necessitar de uma divisão da página pai caso esta também já esteja cheia e isso sucessivamente até chegar à página raiz. Se a página raiz também já estiver cheia e tiver que dividir em duas, então cria-se uma nova raiz. Desse modo a B-árvore aumenta de altura.

Veremos um exemplo completo inserindo chaves em uma B-árvore de ordem 2 inicialmente vazia.

# Exemplo de inserção: inserir 20

Exemplo de inserção: inserir 20 numa B-árvore de ordem 2 inicialmente vazia

20

# Exemplo de inserção: inserir 40

20 40

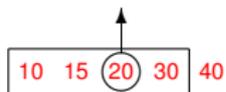
# Exemplo de inserção: inserir 10

10 20 40

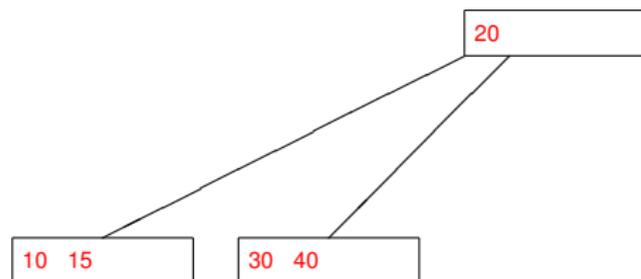
# Exemplo de inserção: inserir 30

10 20 30 40

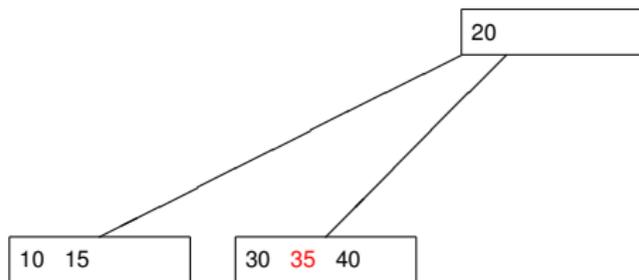
# Exemplo de inserção: inserir 15 - excedeu capacidade



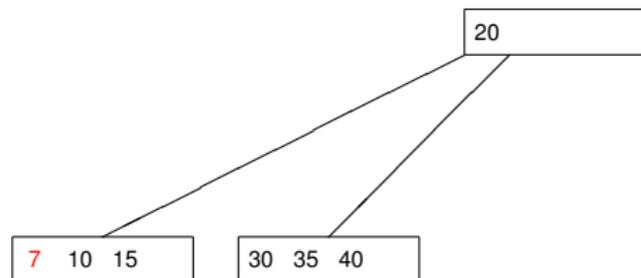
# Exemplo de inserção: divisão de página



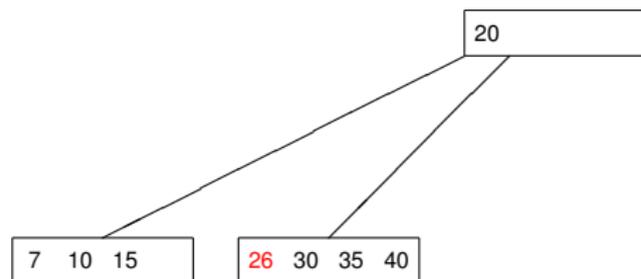
# Exemplo de inserção: inserir 35



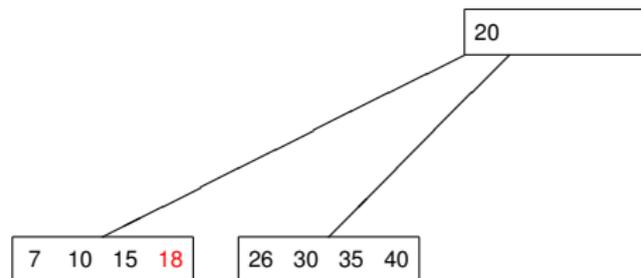
# Exemplo de inserção: inserir 7



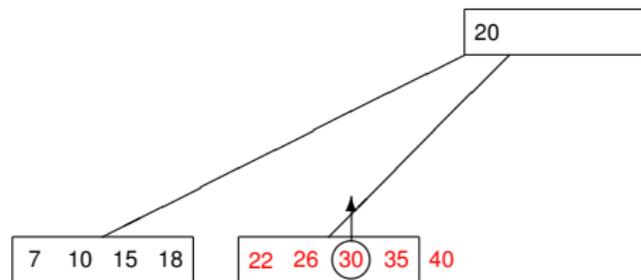
# Exemplo de inserção: inserir 26



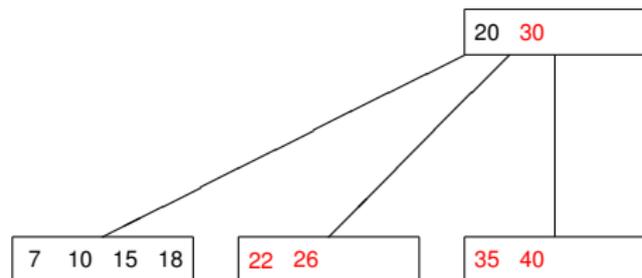
# Exemplo de inserção: inserir 18



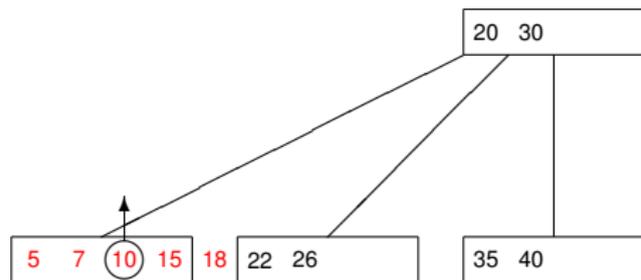
# Exemplo de inserção: inserir 22 - excedeu capacidade



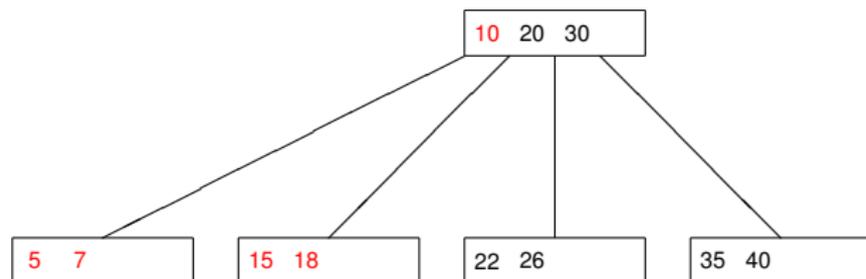
# Exemplo de inserção: divisão de página



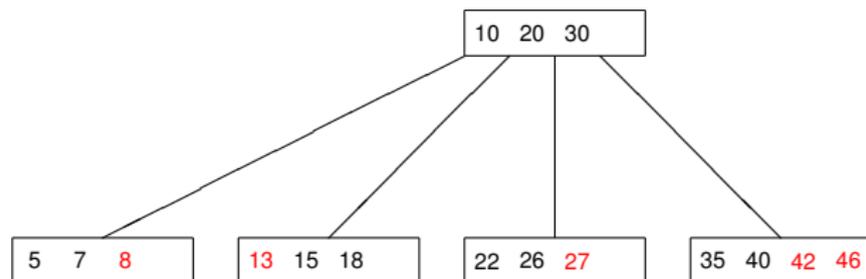
# Exemplo de inserção: inserir 5 - excedeu capacidade



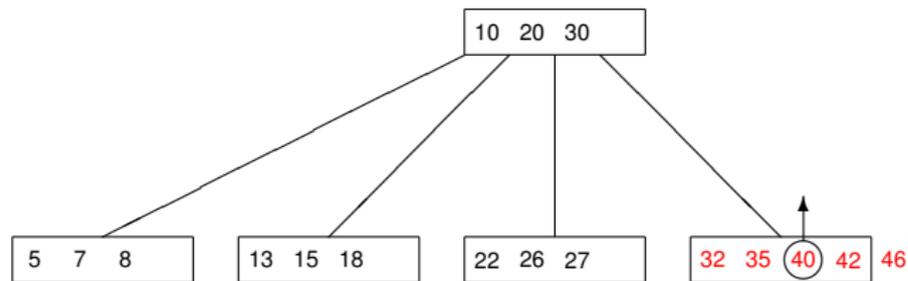
# Exemplo de inserção: divisão de página



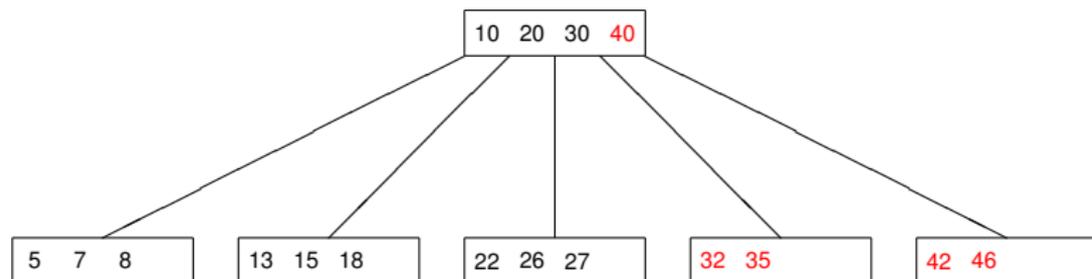
# Exemplo de inserção: inserir 42, 13, 46, 27, 8



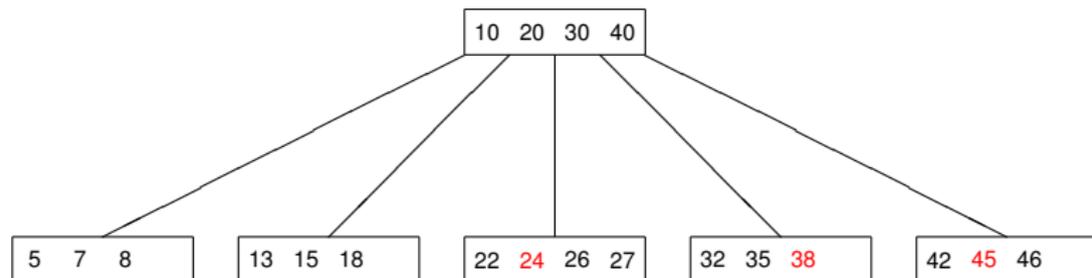
# Exemplo de inserção: inserir 32 - excedeu capacidade



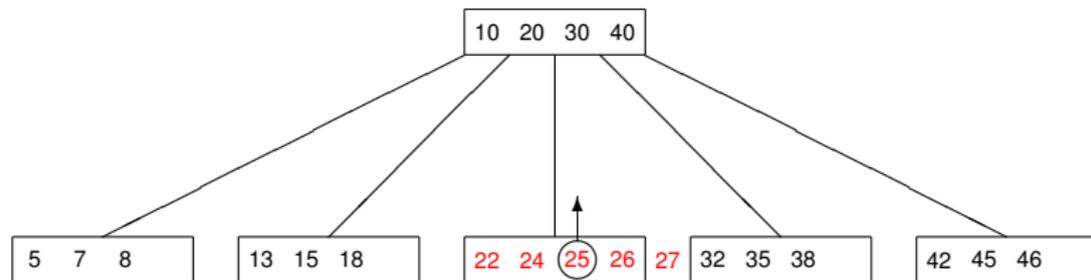
# Exemplo de inserção: divisão de página



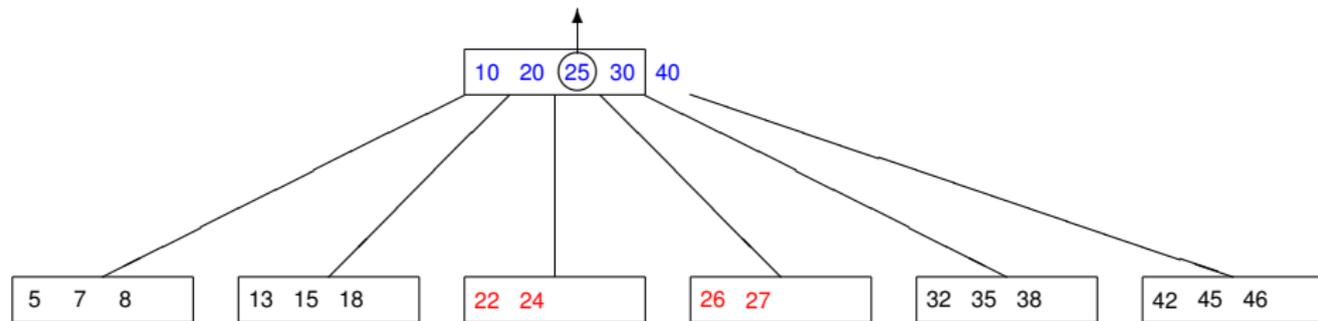
# Exemplo de inserção: inserir 38, 24, 45



# Exemplo de inserção: inserir 25 - excedeu capacidade

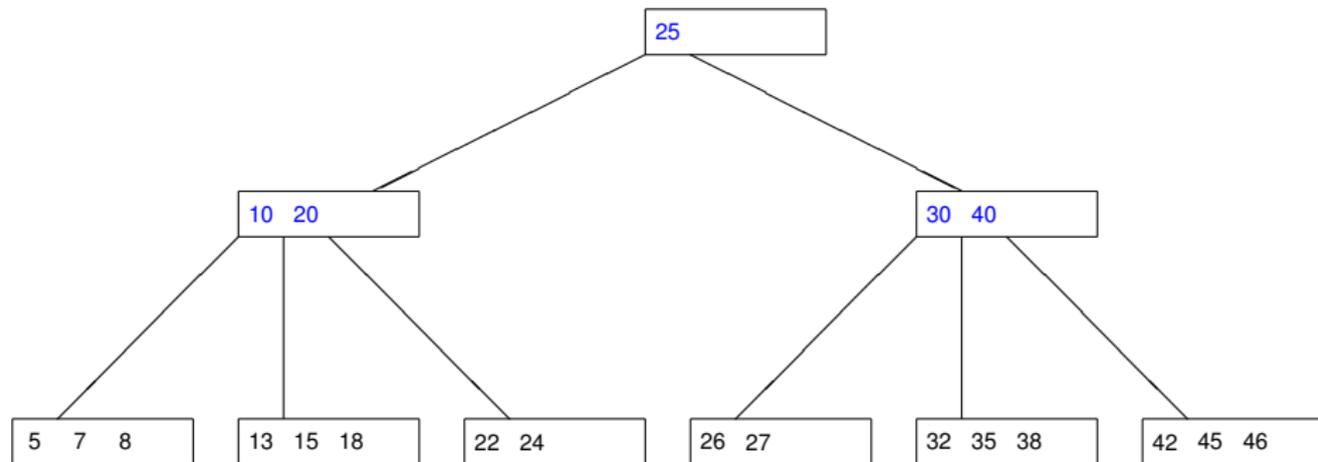


# Divisão de página - estourou na página acima



# Divisão da raiz - criação de nova raiz

A B-árvore aumentou de altura



# Remoção da B-árvore de ordem $b$

Em uma B-árvore de ordem  $b$  cada nó (exceto a raiz) contém entre  $b$  a  $2b$  chaves. A raiz pode conter entre 1 a  $2b$  chaves.

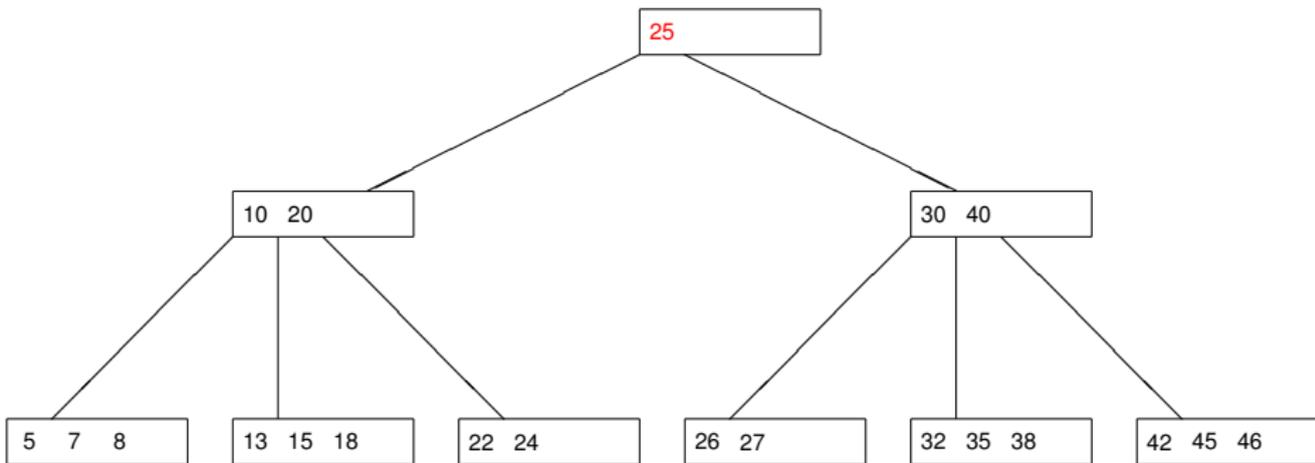
Para remover uma chave  $x$  de uma B-árvore de ordem  $b$ :

- Primeiro localizamos a página  $P$  onde fica a chave  $x$ .  $P$  pode ser uma página folha ou uma página interna.
- Caso 1:  $P$  é uma página folha. Retira-se a chave  $x$ . Se após retirar a chave  $x$ ,  $P$  contém menos de  $b$  chaves, então reparamos da seguinte forma.
  - Caso 1.1: O número total de chaves de  $P$  e de uma página irmã  $Q$  é  $\geq 2b$ : então fazemos uma **redistribuição** de chaves de  $Q$  em direção à  $P$ , mantendo as propriedades de B-árvore.

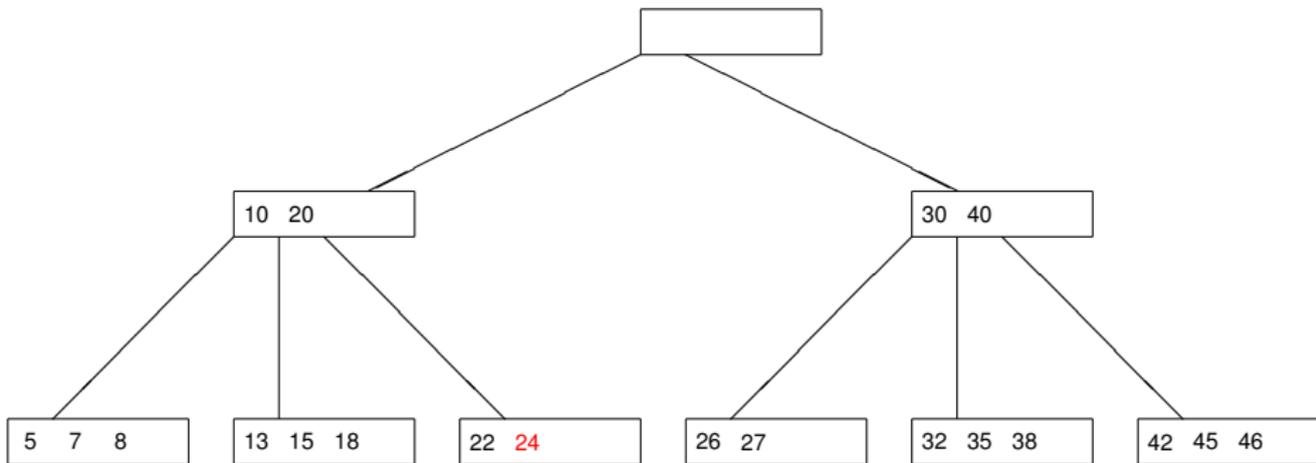
# Remoção da B-árvore de ordem $b$

- Caso 1:  $P$  é uma página folha. Retira-se a chave  $x$ . Se após retirar a chave  $x$ ,  $P$  contém menos de  $b$  chaves, então reparamos da seguinte forma.
  - Caso 1.2: O número total de chaves de  $P$  e de uma página irmã  $Q$  é  $< 2b$ : então fazemos uma **concatenação** das páginas  $P$  e  $Q$ , juntando as suas chaves, mais uma chave da página pai. A página pai tem uma chave retirada, podendo passar a conter menos de  $b$  chaves. O processo de redistribuição ou concatenação precisa ser repetido, recursivamente, até o pior caso de se chegar ao topo da B-árvore, quando a raiz original desaparece. É dessa forma a B-árvore diminui de altura.
- Caso 2:  $P$  é uma página interna. Substitui-se  $x$  pela chave  $y$  de maior valor da subárvore esquerda. Essa chave  $y$  é a chave mais à direita na página mais à direita da subárvore esquerda de  $x$ . A chave  $y$  se localiza numa página folha. A retirada de  $y$  de uma folha faz recair no caso 1.

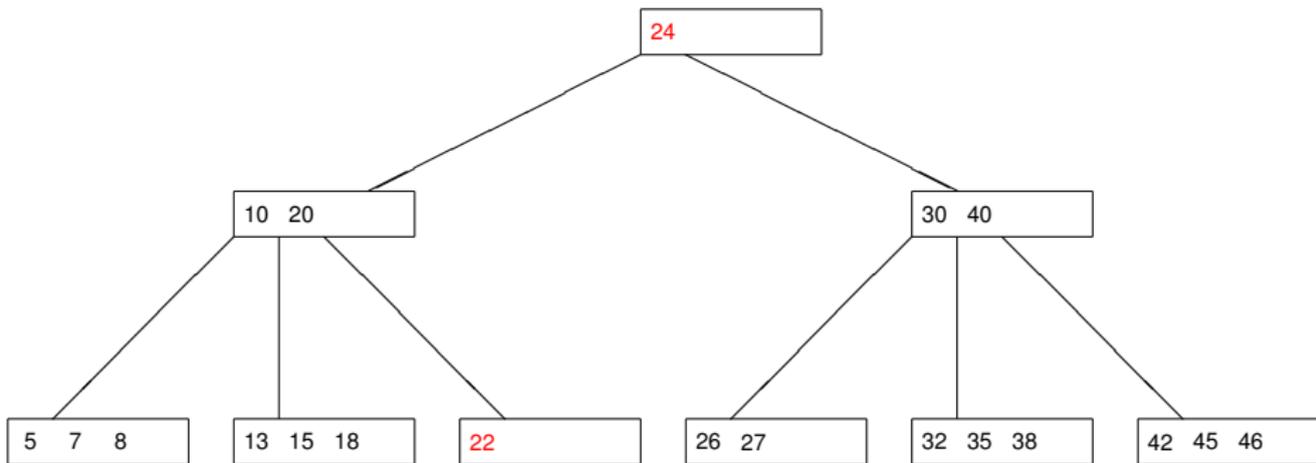
# Exemplo de remoção - remover 25



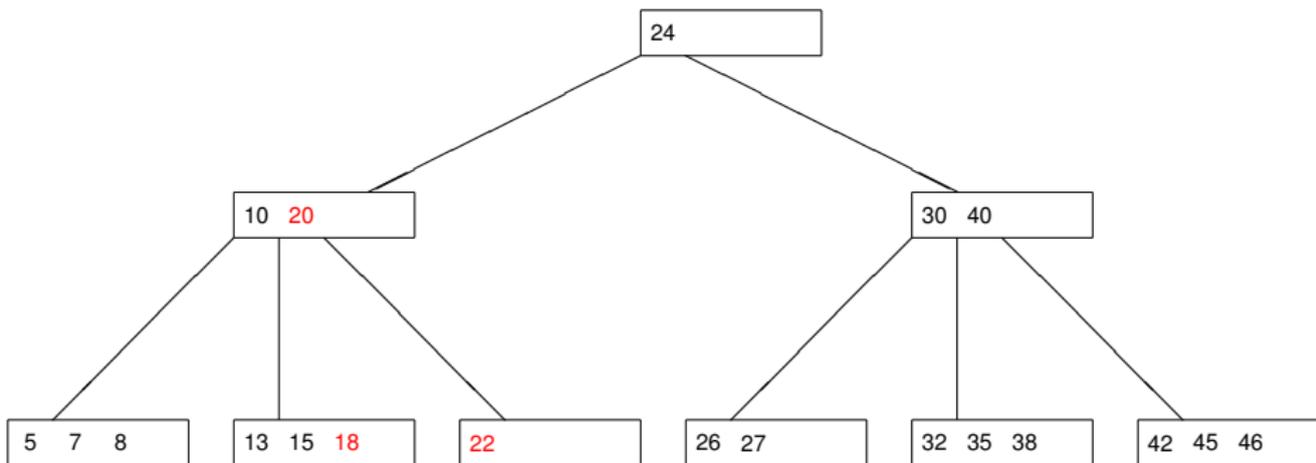
# Substituir pela maior chave da subárvore esquerda 24



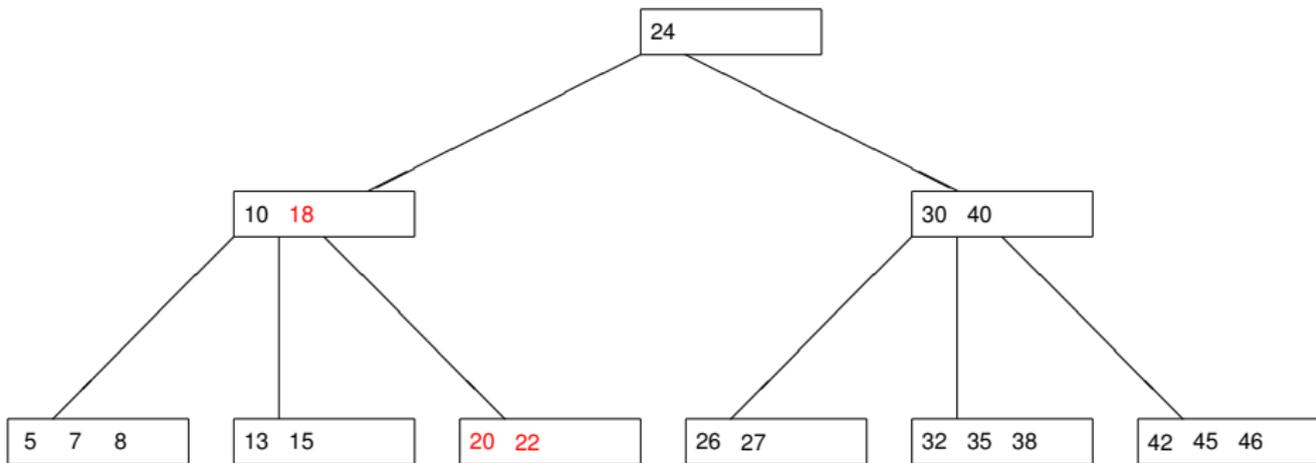
# Agora a página com chave 22 tem poucas chaves



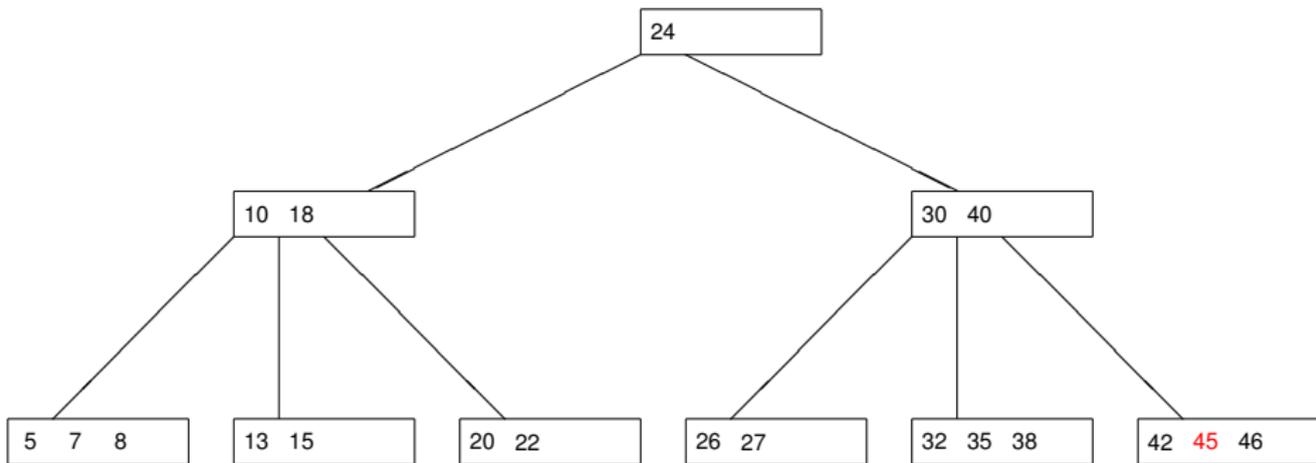
# Redistribuir as chaves com a página irmã



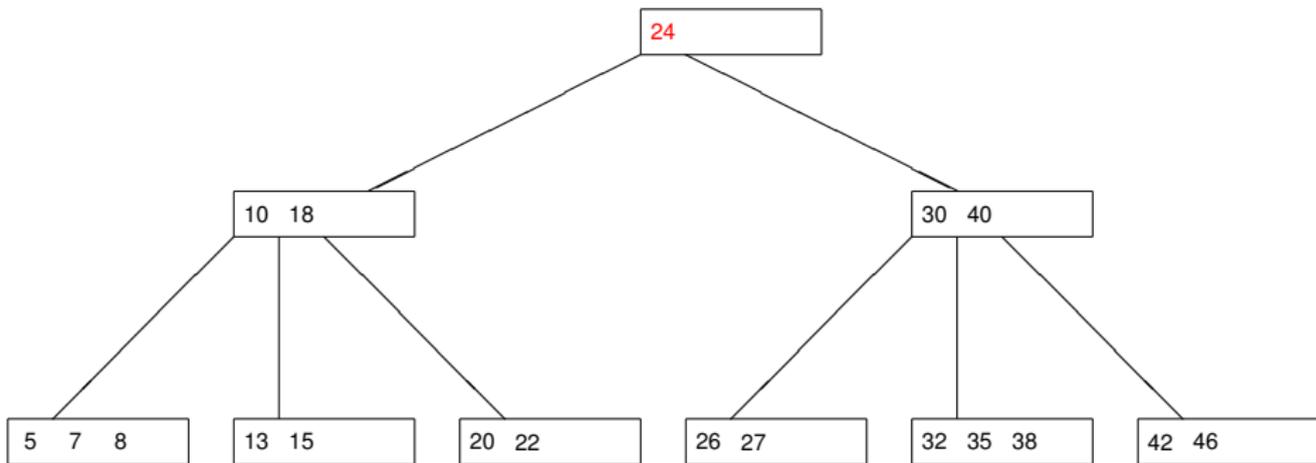
# Agora a página tem número suficiente de chaves



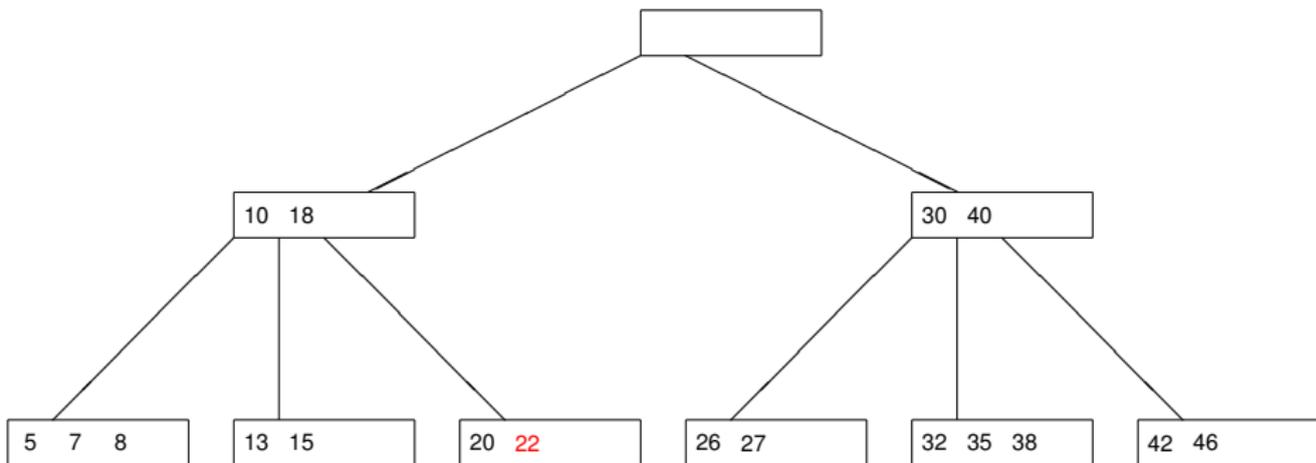
# Exemplo de remoção - remover 45



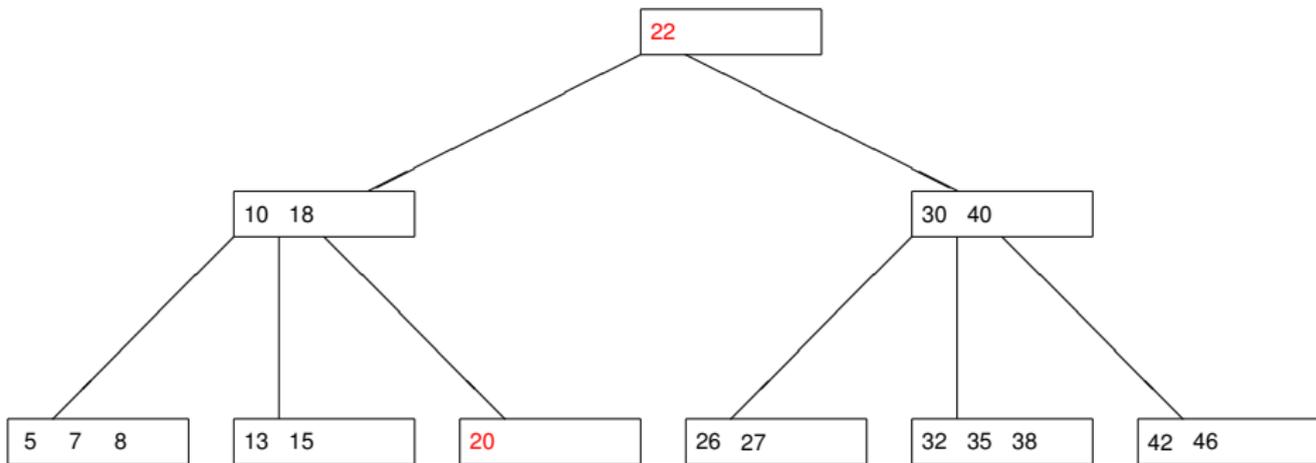
# Exemplo de remoção - agora remover 24



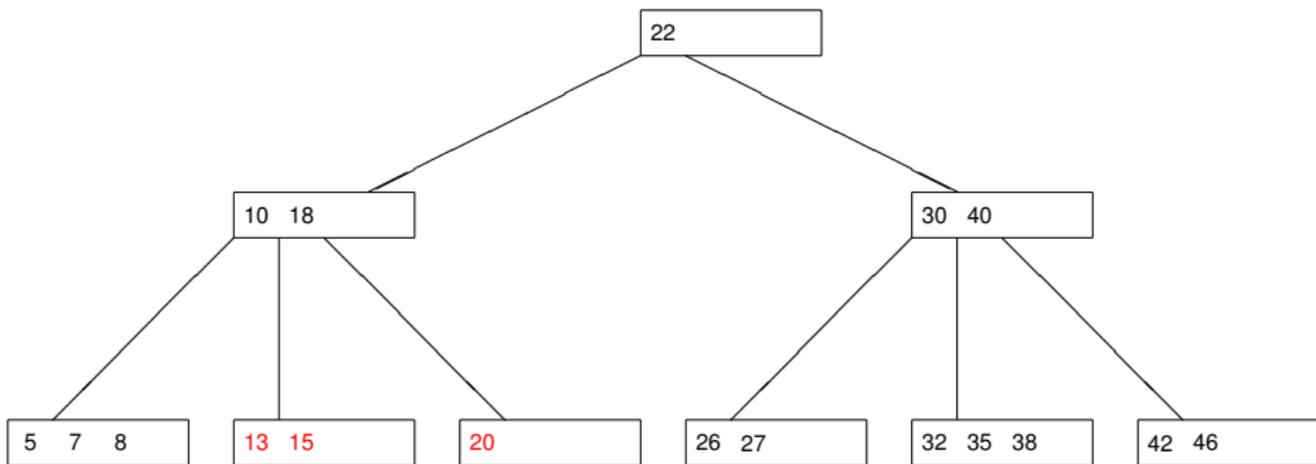
# Substituir pela maior chave da subárvore esquerda 22



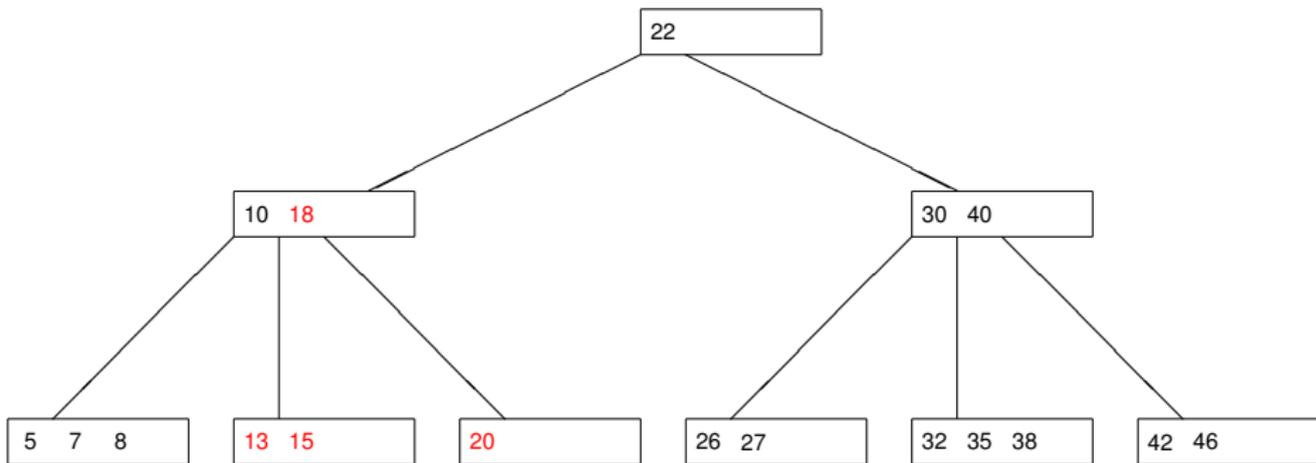
# Agora a página com chave 20 tem poucas chaves



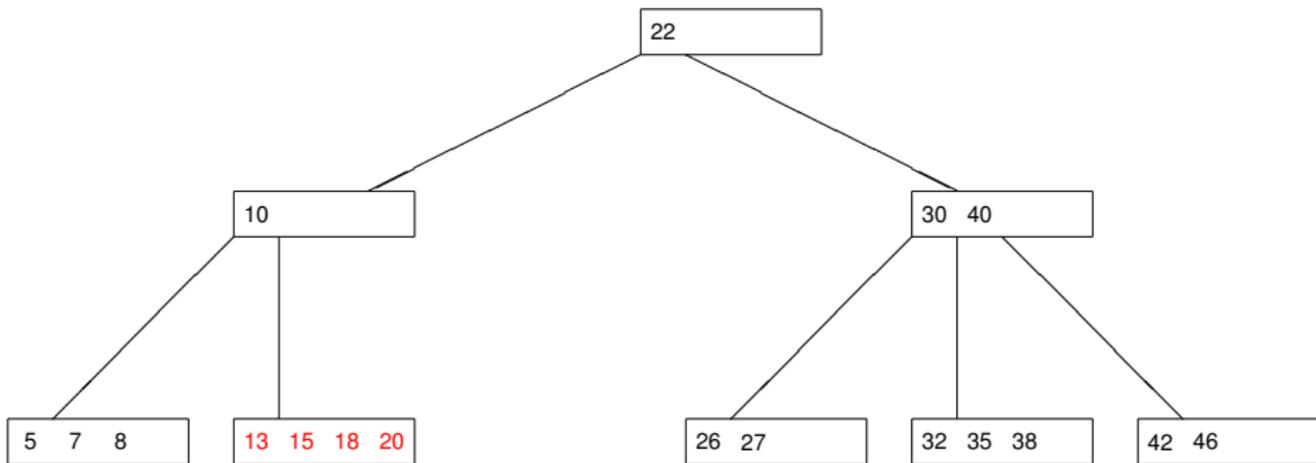
# A página irmã tem poucas chaves para redistribuir



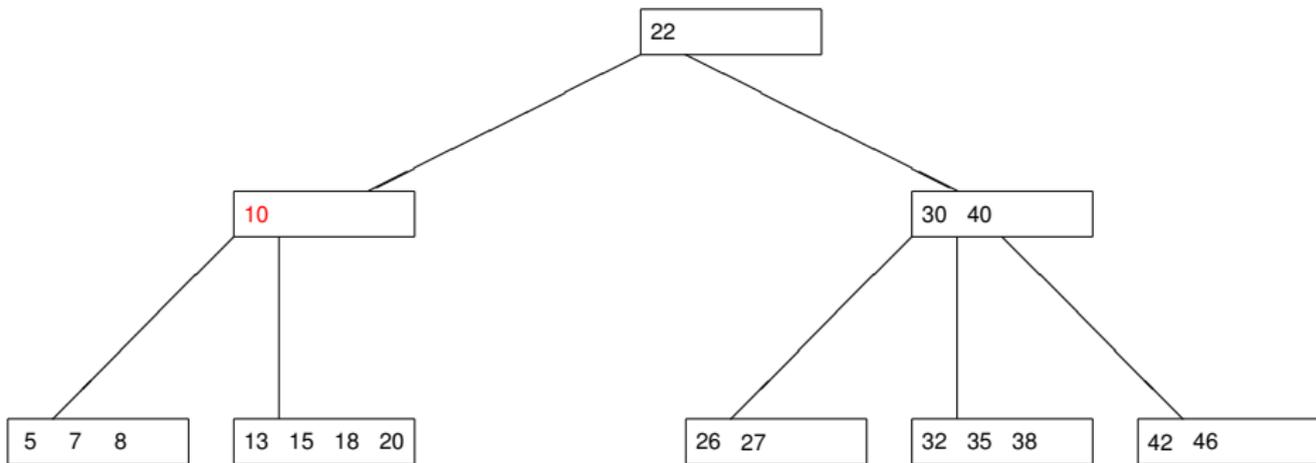
# Então concatenar com as chaves da página irmã



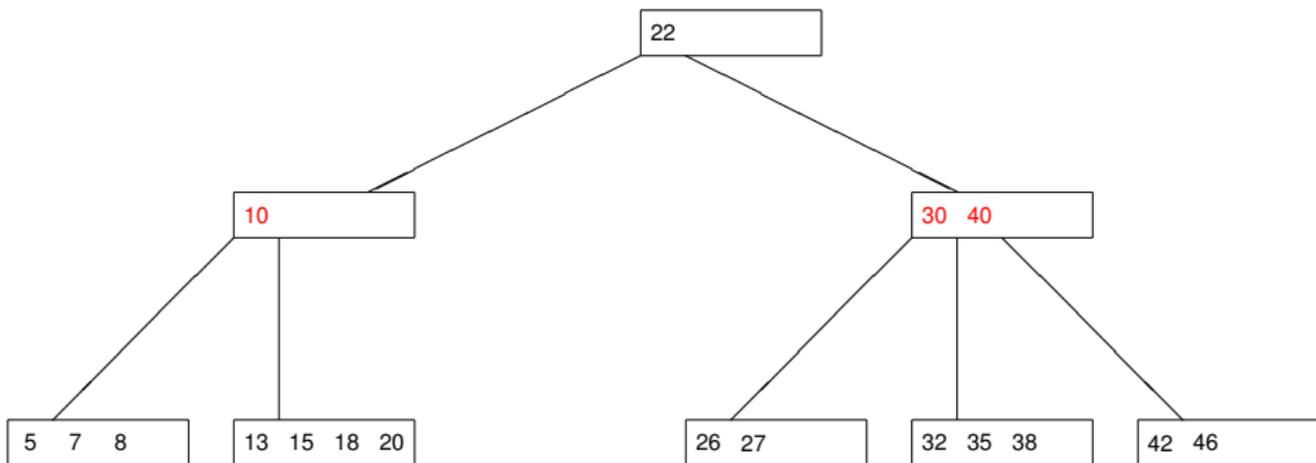
# Resolveu o problema, mas ...



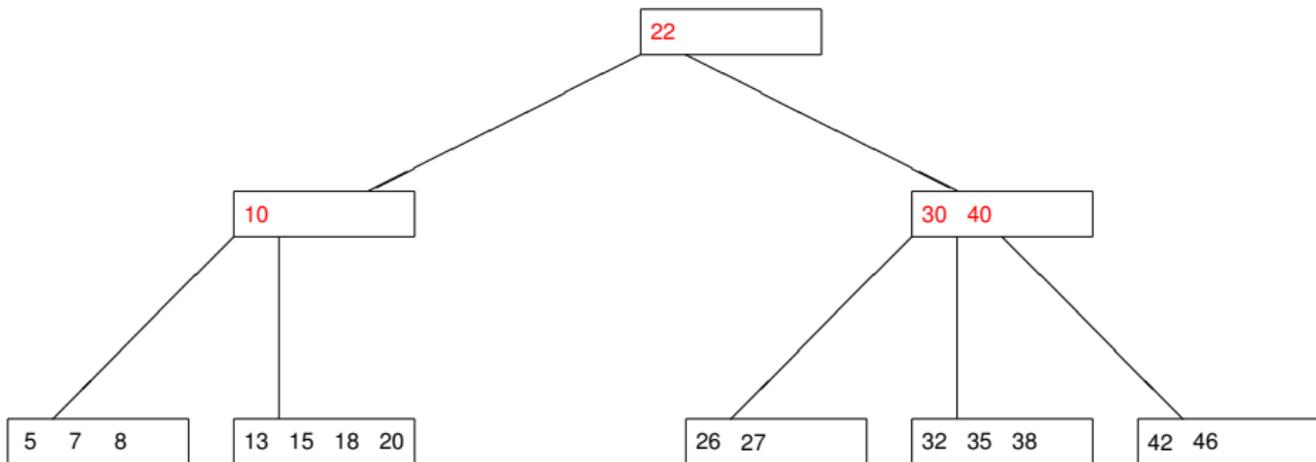
# Agora a página com a chave 10 tem poucas chaves



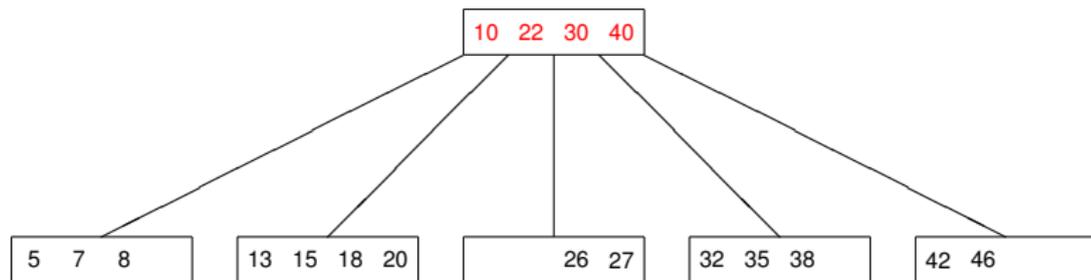
# Não dá para redistribuir chaves da página irmã



# Então vamos concatenar as chaves



# A B-árvore diminuiu de altura



# Exercício sobre remoção

Considere a B-árvore do slide anterior.

- Desenhe a nova B-árvore depois de remover cada uma das seguintes chaves: 38, 32, 8, 27, 46, 13, 42, 5, 22, 18, 26, 7, 35, 15.
- Dica: Após remover a última chave acima (15), a B-árvore que resulta deve ter uma página apenas, contendo as chaves 10, 20, 30 e 40.