

# Árvores B

## □ Motivação: pesquisa em disco

- Tempo de acesso a disco determinante nas operações
  - Com disco de 10 ms de tempo de acesso:  
100 acessos por segundo  
em máquina de 25 MIPS, 1 acesso custa tanto como 250 000 instruções  
... e velocidade de cálculo tem aumentado mais que velocidade de disco
  - Podemos gastar muito cálculo para evitar 1 acesso a disco
- Exemplo: registo dos cidadãos portugueses guardado em árvore binária
  - $N=10M$  de registos, chave de 32 bytes e total de registo de 256 bytes (BD 2.5GB)
  - Blocos de 8KB  $\rightarrow$  312 500 blocos
  - Pior caso: acesso em tempo linear, 312 500 acessos a disco (52 min) (ou 10M !)
  - Médio:  $1.38 \log N$  acessos, 32 acessos a disco ou 0.3s
  - Nó a profundidade tripla:  $\approx 100$  acessos a disco ou 1s

B-tree

# Árvore B (ou melhor B+)

## □ Árvore equilibrada multivia

- Objectivo: pesquisa externa, minimizar acessos a disco
- Utilização: construir índices auxiliares em Bases de Dados
  - nos nós existem referências a blocos do disco
  - escolha da ordem da árvore depende do número de registos do índice que cabem num único bloco (unidade básica de acesso a disco)

Uma árvore-B de ordem  $M$  é uma árvore de  $M$  vias em que

1. Os dados estão guardados nas folhas
2. Os nós internos guardam até  $M-1$  chaves; a chave  $i$  representa a menor chave na subárvore  $i+1$
3. A raiz é uma folha ou tem entre 2 e  $M$  filhos
4. Todos os nós internos, excepto a raiz, têm entre  $\lceil M/2 \rceil$  e  $M$  filhos não vazios.
5. Todas as folhas estão à mesma profundidade e têm entre  $\lceil L/2 \rceil$  e  $L$  registos de dados

B-tree

# Árvore B para exemplo do ficheiro

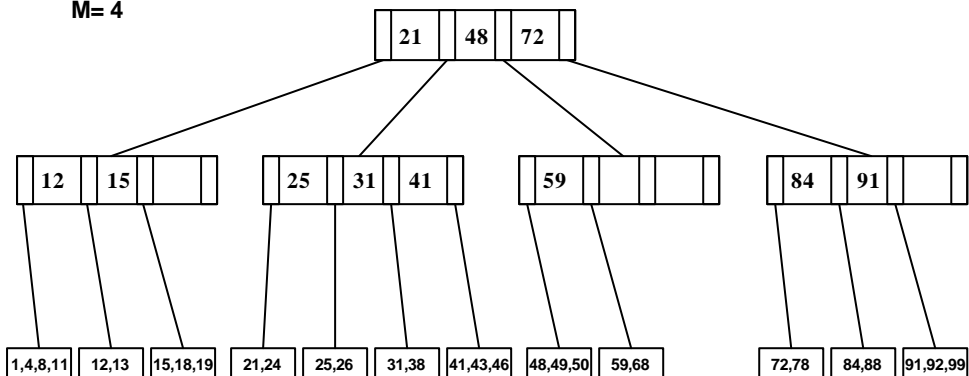
## □ Cada nó deve caber em 1 bloco de disco

- Assumir bloco de disco de 8 192 bytes
- Em cada nó interior:
  - M-1 chaves de 32 bytes e M ramos com 4 bytes (endereço de novo bloco)
  - $32 (M - 1) + 4 M \leq 8192$
  - $M = 228, M/2 = 114$
- Nos nós folha:
  - 256 bytes para cada registo
  - bloco comporta 32 registos
  - $L = 32$
- No ficheiro de 10 000 000 de registos
  - no máximo  $10\,000\,000 / 16 = 625\,000$  folhas
  - no máximo  $\lceil \log_{114} 625\,000 \rceil = 3$  níveis acima das folhas

B-tree

## Árvore-B de ordem 4

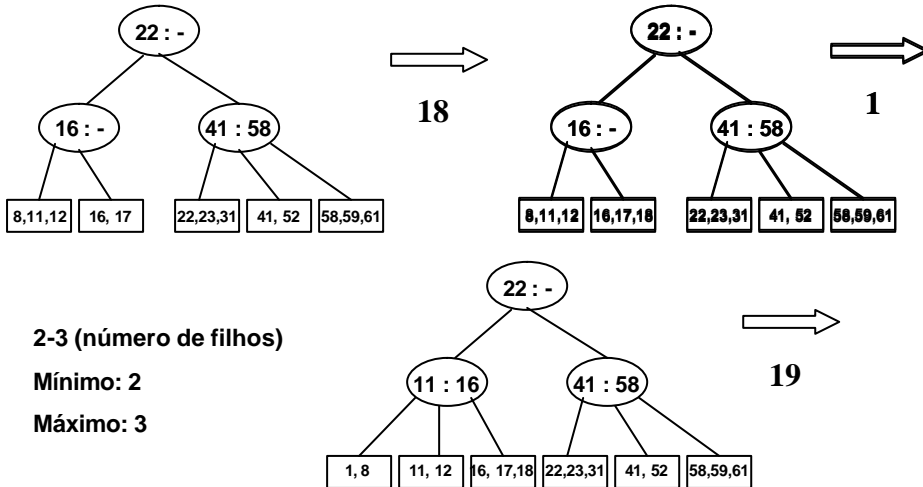
M= 4



- até 3 chaves em cada nó interno
- até 4 ponteiros em cada nó interno (mínimo é 2)
- $L = 4$  (mas poderia ter outro valor)

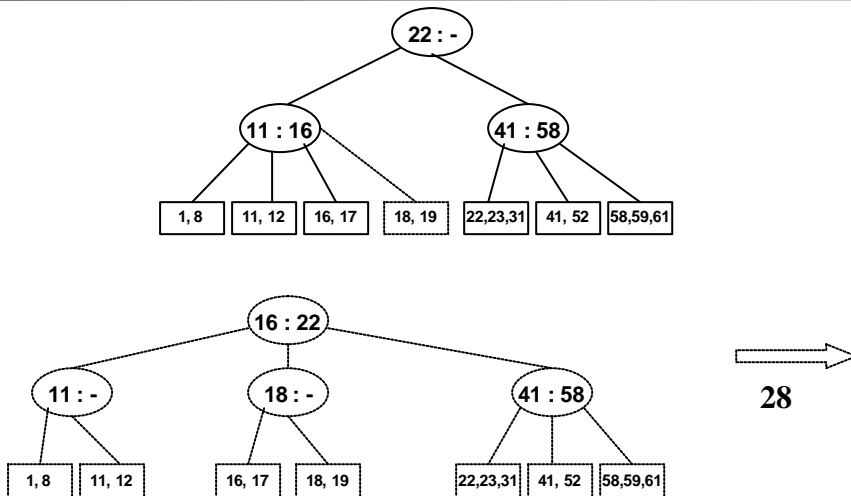
B-tree

## Inserção em Árvore-B 2-3



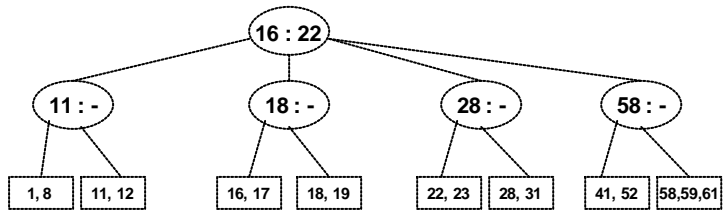
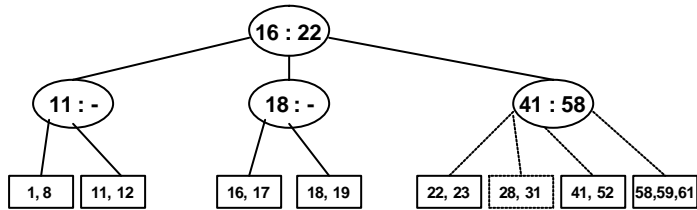
B-tree

## Inserção em Árvore-B 2-3



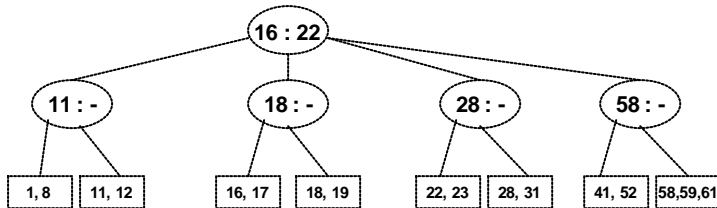
B-tree

## Inserção em Árvore-B 2-3

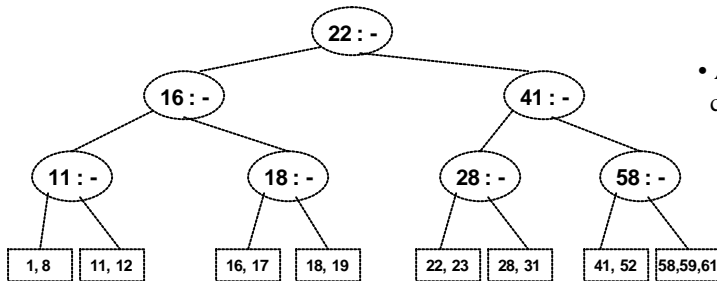


B-tree

## Inserção em Árvore-B 2-3



(repetida)



- Árvore-B cresce para a raiz

B-tree

# Apagamento em Árvore B

## ❑ Pesquisa da chave a apagar e apagamento na folha respectiva

- Se a folha fica com número de chaves não abaixo do mínimo: terminar
- Se a folha fica com número de chaves abaixo do mínimo: reparar árvore
  - Se a folha do lado tiver número de chaves acima do mínimo: pedir chave emprestada
    - “do lado” significa com o mesmo pai, para não complicar excessivamente a manutenção dos nós interiores
  - Se a folha do lado não tiver número de chaves acima do mínimo: fundir folhas e propagar
- Havendo fusão de nós, a reparação prossegue nos níveis superiores da árvore
- Se a fusão de nós resulta em a raiz ter apenas 1 filho: raiz passa para o filho e altura da árvore diminui

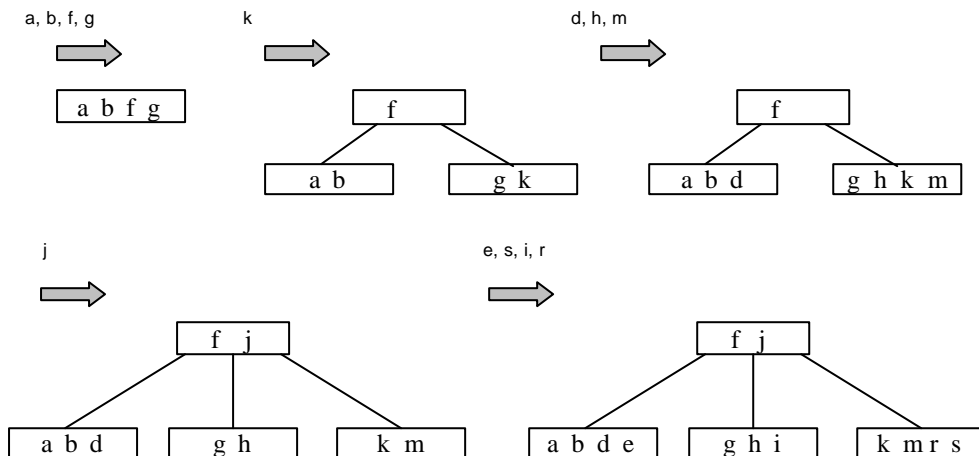
## ❑ Árvore-B cresce e diminui na raiz

- altura da árvore aumenta quando se tem de partir a raiz (e criar novo nó raiz)
- altura da árvore diminui quando a raiz tem 2 filhos e estes se fundem

B-tree

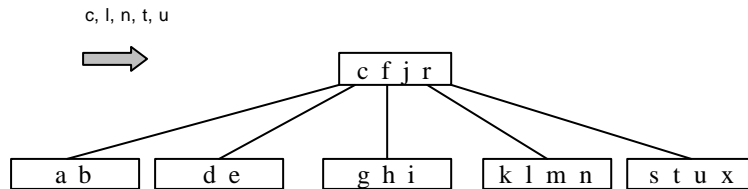
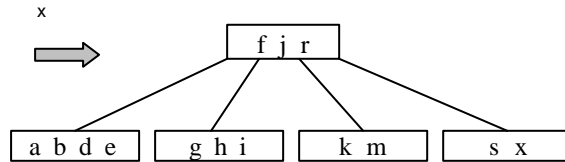
## Árvore-B para pesquisa em memória (não B+)

Árvore de ordem 5 ou B 3-5



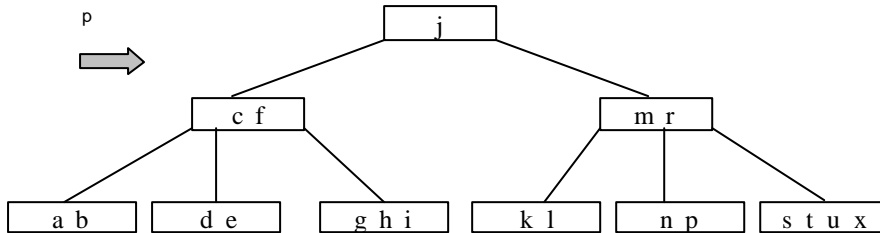
B-tree

## Inserção em árvore-B 3-5



B-tree

## Inserção em árvore-B 3-5



- Nesta representação:
- chaves estão organizadas na árvore à maneira das árvores de pesquisa
- nós internos e folhas são da mesma natureza

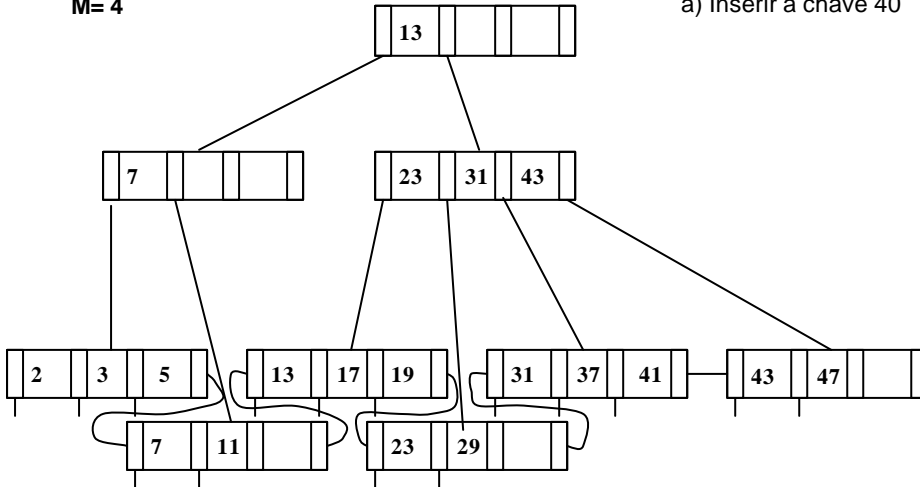
Apropriado para estruturas em memória central

B-tree

## Exercício (árvore B+ com folhas ligadas sequencialmente)

M= 4

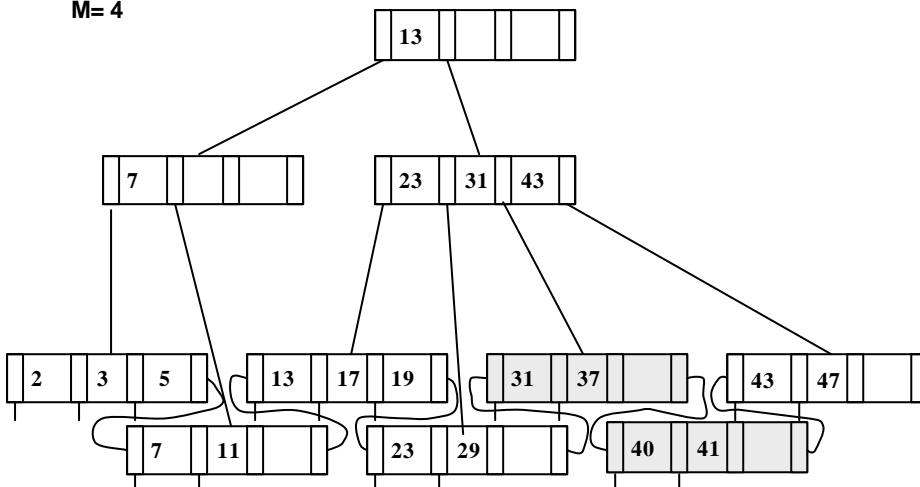
a) Inserir a chave 40



B-tree

## Exercício (cont.)

M= 4

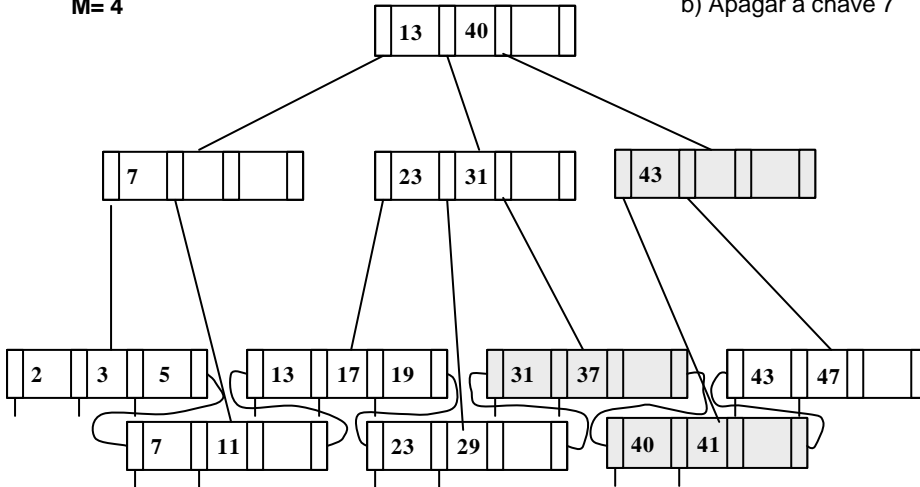


B-tree

## Exercício (cont.)

M= 4

b) Apagar a chave 7

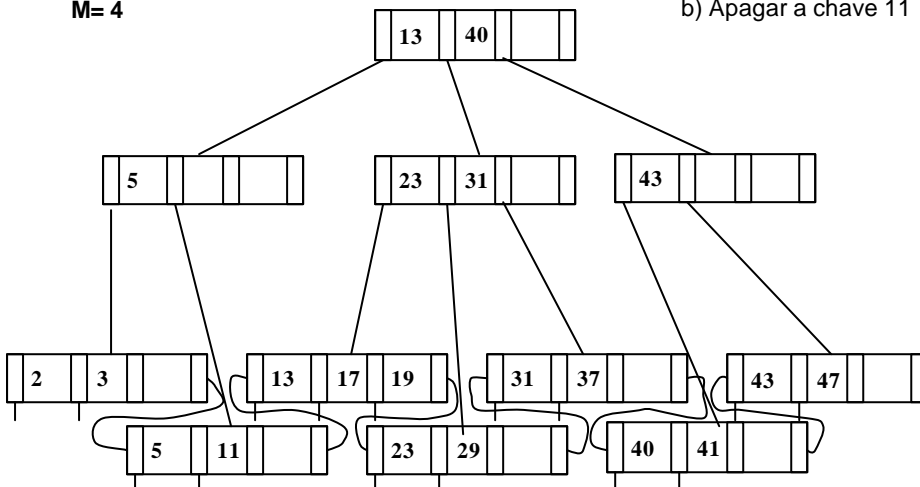


B-tree

## Exercício (cont.)

M= 4

b) Apagar a chave 11

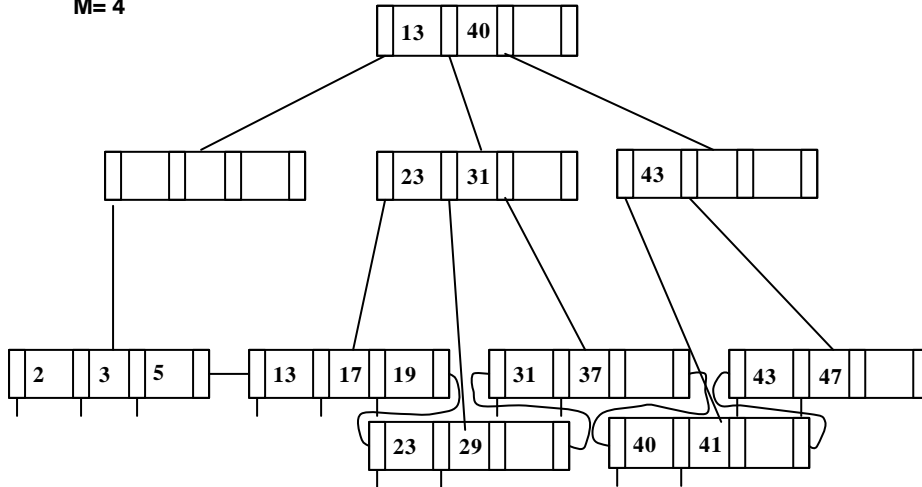


B-tree



## Exercício (cont.)

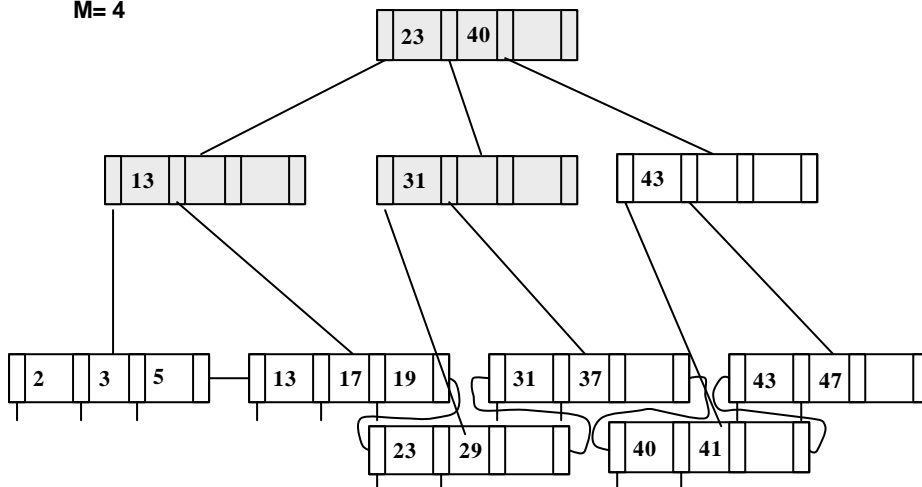
M= 4



B-tree

## Exercício (fim)

M= 4



B-tree