

-
-
-
-
-
-
-

Árvores B

Algoritmos e Estruturas de Dados

2020/2021



FEUP

-
-
-
-
-
-
-
-

Árvores B

- Nas estruturas de dados anteriores:
 - todos os dados residem em memória
- E se a memória principal for insuficiente?
 - necessário recorrer a memórias secundárias ... discos
- Que estruturas de dados para guardar dados em disco?
 - vetores ordenados em ficheiro
 - inserção de elementos exige movimentação de dados no ficheiro
 - árvores em ficheiros
 - não exigem movimentação de elementos
 - um bloco de disco corresponde a um nó da árvore
 - como reduzir o número de acessos?
 - árvore AVL melhor que BST
 - porque não aumentar o número de filhos de um nó, e assim diminuir a altura?

Árvores B

- árvore binária completa tem altura $\approx \log_2 N$
- árvore de ordem M completa tem altura $\approx \log_M N$
- **Árvore B de ordem M**
 - os dados estão guardados nas folhas
 - os nós internos guardam até $M-1$ chaves; a chave i representa a menor chave na sub-árvore $i+1$
 - A raiz é uma folha ou tem entre 2 e M filhos
 - Todos os nós internos, exceto a raiz, têm entre $M/2$ e M filhos não vazios
 - Todas as folhas estão à mesma profundidade e têm entre $L/2$ e L filhos
 - como determinar os valores de M e L ?

Árvores B

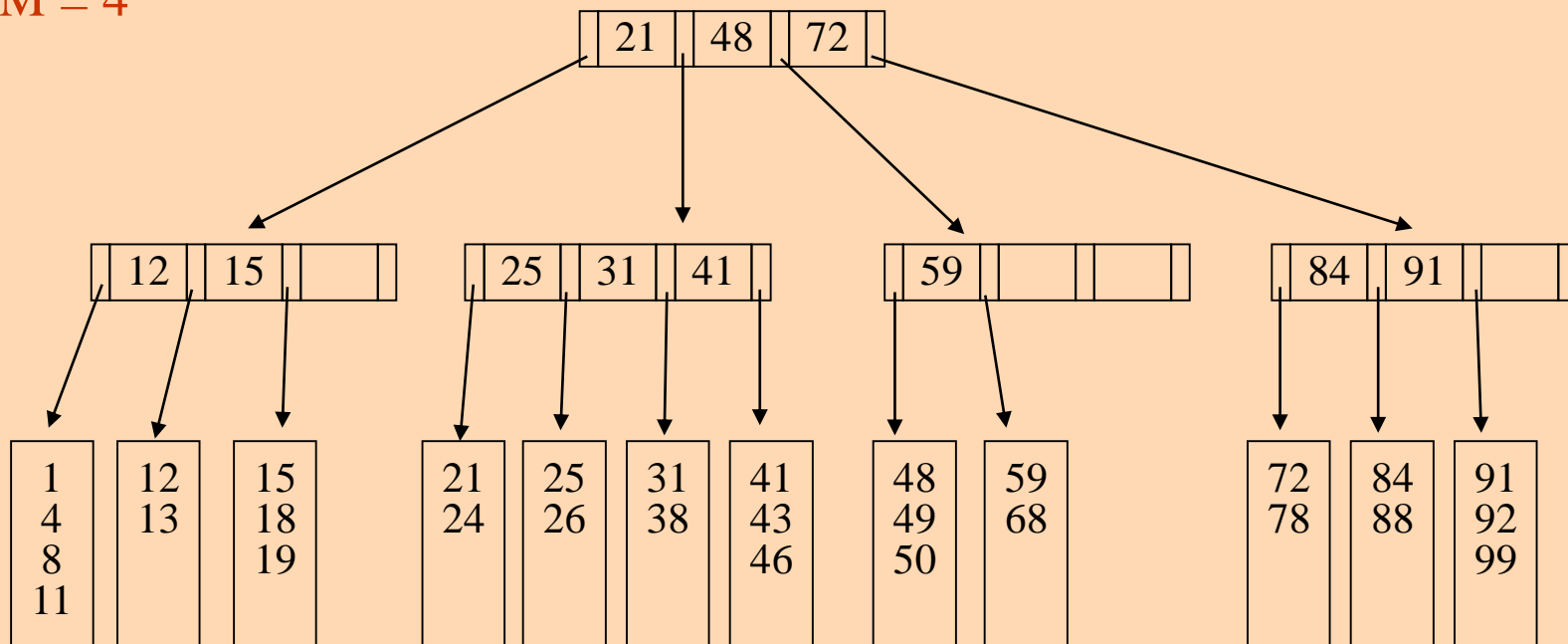
exemplo: registo de cidadãos

- existem 10 000 000 itens; chave de 32 bytes (nome) e registo de 256 bytes
- cada nó representa um 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)
 - $32*(M-1) + 4*M \leq 8192$
 - $M = 228$
- nos nós folha:
 - 256 bytes para cada registo
 - bloco comporta 32 registos
 - $L = 32$



Árvores B

$M = 4$



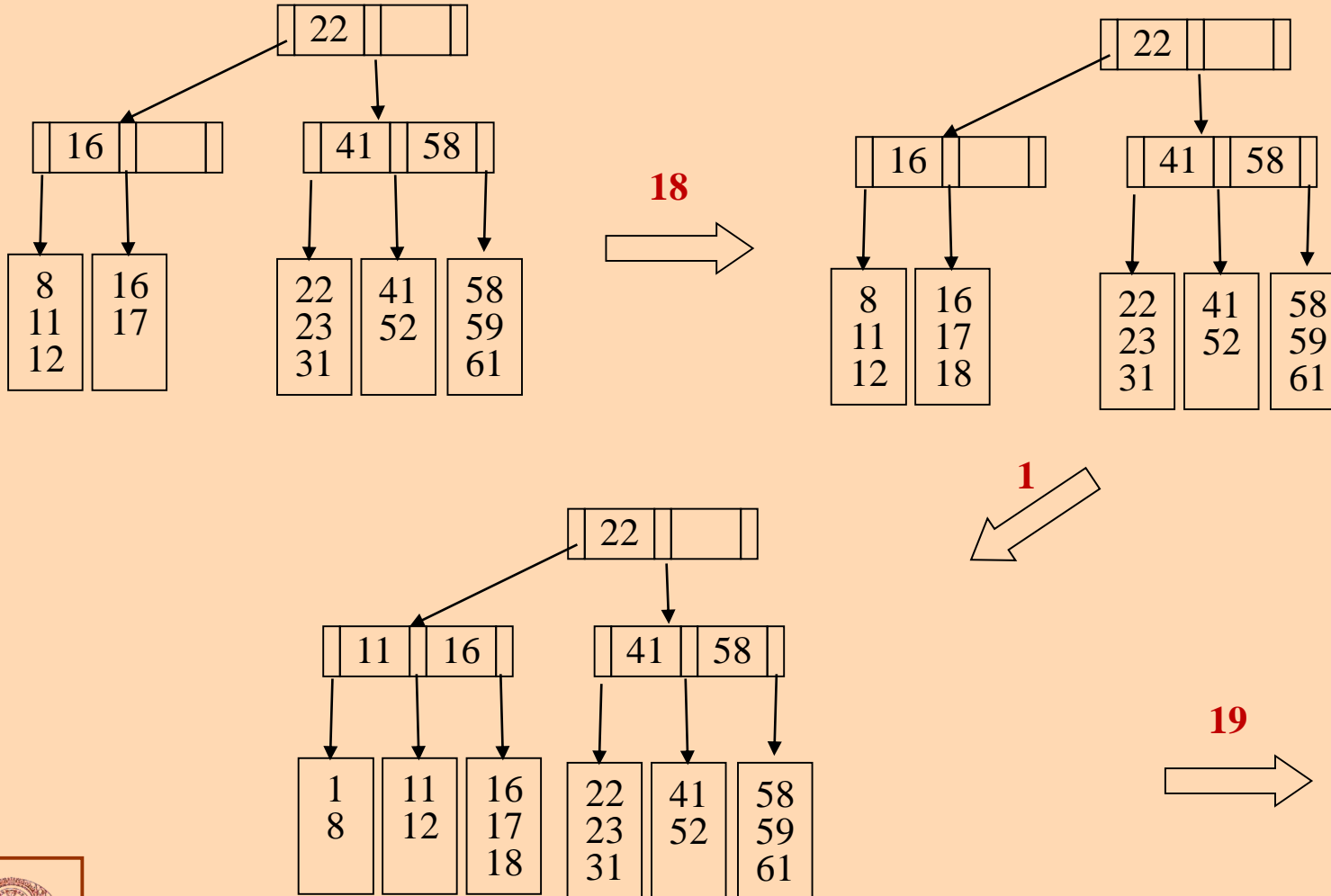
- até 3 chaves em cada nó interno
- cada nó interno possui entre 2 e 4 filhos
- $L = 4$ (neste exemplo) . cada nó folha possui entre 2 e 4 filhos



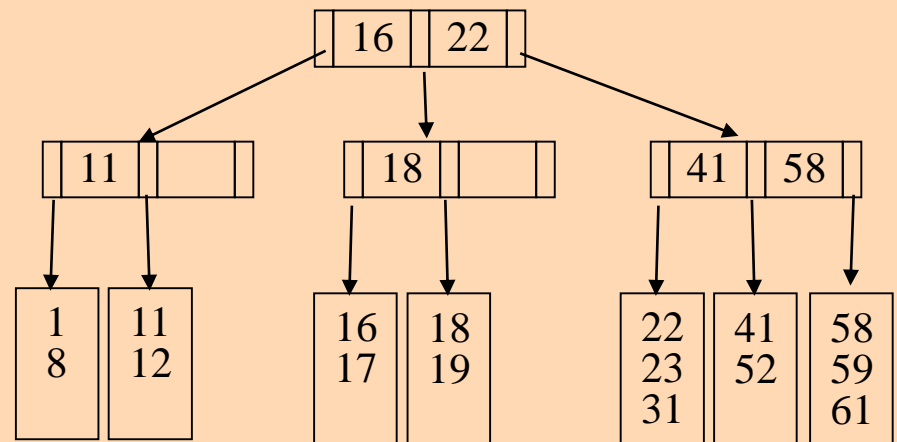
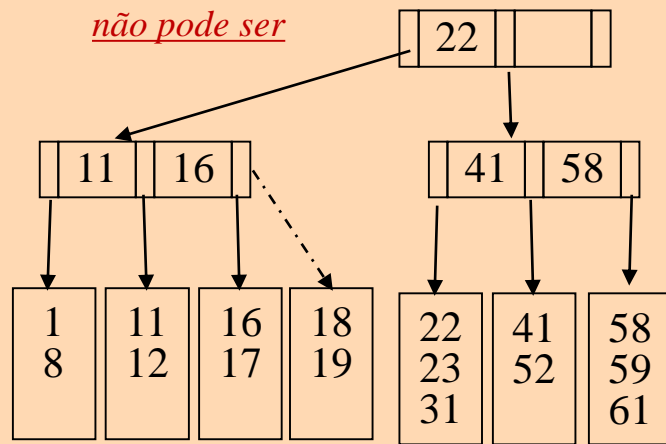
Árvores B

- **Inserir um elemento**
 - pesquisar a chave e inserir na folha respectiva
 - se folha fica com número de chaves não superior ao máximo, terminar
 - se folha fica com número de chaves superior ao máximo: reparar árvore
 - dividir em duas folhas; alterar chaves e filhos do pai
 - se pai fica com número de filhos superior ao máximo, propagar a separação para o pai
 - Se a separação ocorre na raiz, criar uma nova raiz com os nós separados como filhos (*árvore sobe*)

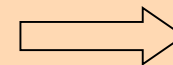
Inserção em Árvore B 2-3



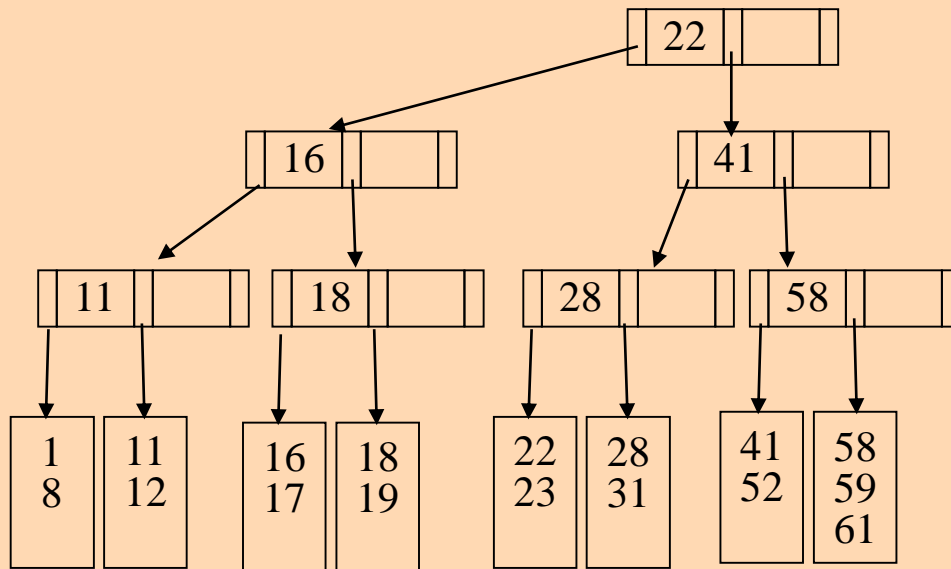
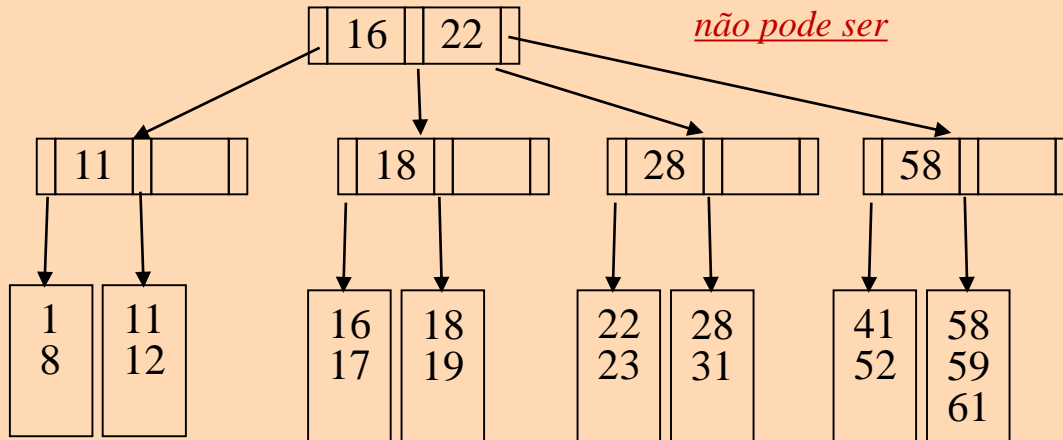
Inserção em Árvore B 2-3



28



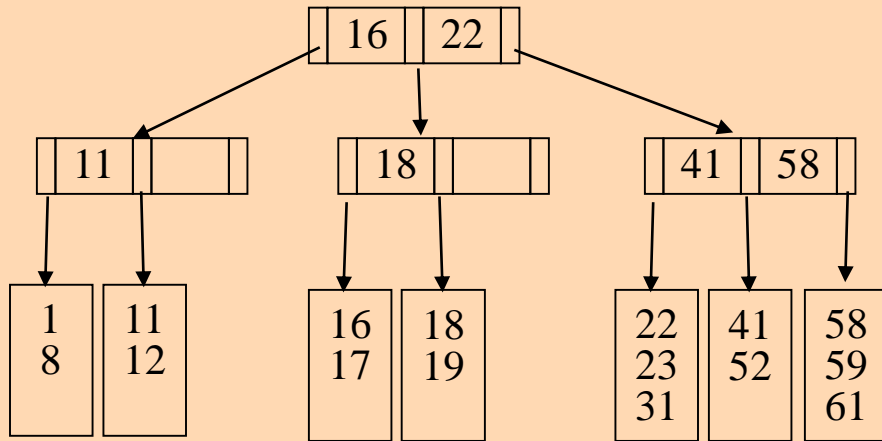
Inserção em Árvore B 2-3



Árvores B

- **Apagar um elemento**
 - pesquisar a chave e apagar na folha respectiva
 - se folha fica com número de chaves não inferior ao mínimo, terminar
 - se folha fica com número de chaves inferior ao mínimo: reparar árvore
 - se folha vizinha tiver número de chaves acima do mínimo, pedir chave emprestada
 - senão, fundir chaves com essa folha vizinha e propagar
 - havendo fusão de nós, a reparação prossegue nos níveis superiores
 - se a fusão de nós resulta em raiz possuir apenas um filho, remover a raiz, e tornar o seu filho a nova raiz (*árvore desce*)

Remoção em Árvore B 2-3



remove 19

