

Árvores B – Parte II

Construção Bottom-Up

Adaptado dos Originais de:

Leandro C. Cintra

Maria Cristina F. de Oliveira

Árvores B

- Características
 - paginadas
 - balanceadas
 - bottom-up para a criação
 - nós folhas → nó raiz
- Inovação
 - não é necessário construir a árvore a partir da raiz, como é feito para ABBs e AVLs

2

Construção *Bottom-Up*

- Conseqüências
 - não mais se aloca chaves inadequadas na raiz
 - chaves na raiz da árvore “emergem” naturalmente
 - não é mais necessário tratar o problema de desbalanceamento após este ocorrer
 - como rebalancear uma árvore paginada não é mais uma questão a resolver *a posteriori*
 - balanceamento ocorre naturalmente

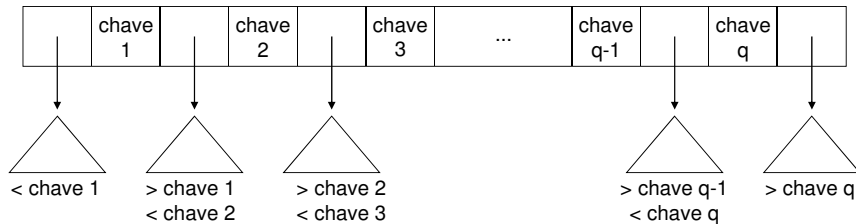
3

Características

- Nó = Página em Disco:
 - Contém uma seqüência de itens ordenados por chave
 - item = registro convencional = (chave, demais informações); **OU**
 - item = registro de índice = (chave, endereço)
 - endereço = RRN ou byte offset do registro no arquivo principal
 - itens de tamanho fixo → páginas com no. fixo de itens
 - Contém um conjunto de ponteiros
 - número de ponteiros = número de chaves + 1
 - Não é internamente encadeado como uma árvore
 - página contém uma lista ordenada seqüencial

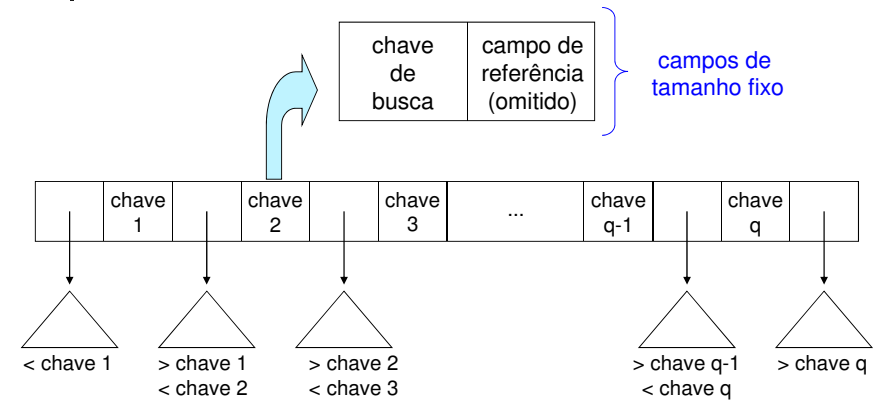
4

Estrutura Lógica de um Nó



5

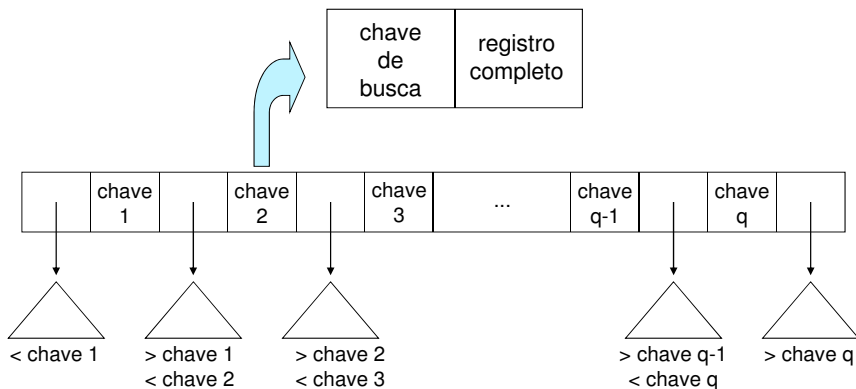
Nó com Registros de Índice



Note que q é fixo: configuração adotada na maior parte das discussões subseqüentes

6

Nó com Registros de Informação

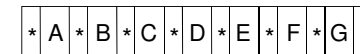


Note que q pode ser variável (se registro for de tamanho variável) ...

7

Características

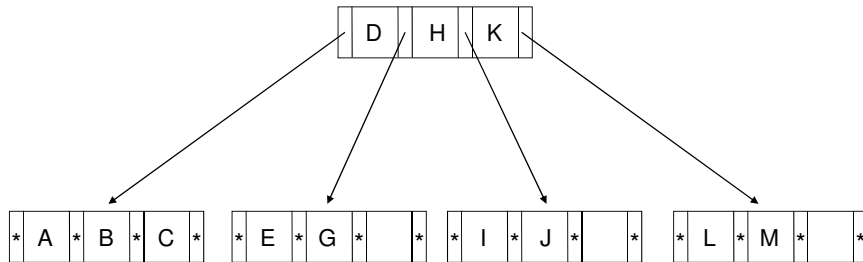
- **Ordem:**
 - No. máx. de ponteiros que podem ser armazenados em um nó
 - Exemplo: árvore B de ordem 8
 - máximo de 7 chaves e 8 ponteiros



- **Observações**
 - No. máx. de ponteiros = no. máx. de descendentes de um nó
 - nós folhas não possuem filhos, e seus ponteiros são nulos

8

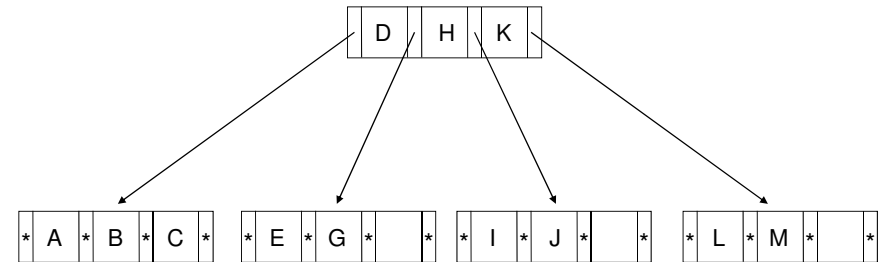
Exemplo de Árvore B (ordem 4)



Por simplicidade, ao longo das discussões subseqüentes destacaremos apenas as chaves dos itens, assim como no exemplo acima

9

Exemplo de Árvore B (ordem 4)



■ Ponteiros:

- Nós folha (*): -1 ou fim de lista (NIL)
- Nós internos: RRN do nó descendente ou -1 se este não existe

10

Inserção de Itens

- Característica:
 - **sempre realizada nos nós folha**
- Situações a Serem Analisadas:
 - árvore vazia (situação inicial)
 - overflow no nó raiz
 - inserção nos nós folhas
 - sem overflow
 - com overflow

11

Inserção: Situação Inicial

- Criação e Preenchimento do 1º Nó:
 - 1ª chave (árvore vazia): criação do nó raiz
 - demais chaves: inserção até capacidade do nó
 - raiz como nó folha
- Exemplo:
 - nó com capacidade para 7 chaves
 - chaves: letras do alfabeto

12

Inserção: Situação Inicial

- Exemplo (cont.):

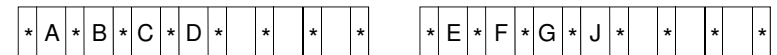
*	A	*	B	*	C	*	D	*	E	*	F	*	G	*
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

 - Chaves B C G E F D A
 - inseridas desordenadamente
 - mantidas ordenadas internamente ao nó
 - a cada chave, página é lida, reordenada em RAM e rescrita
 - Ponteiros (*)
 - -1 ou fim de lista (NIL)
 - nó raiz = nó folha

13

Inserção: Overflow Na Raiz

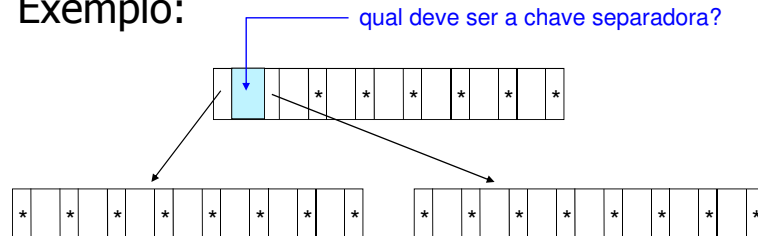
- **Passo 1** – Particionamento do Nó (Split)
 - nó original → nó original + novo nó
 - split "1-to-2" ou "two-way"
 - chaves são distribuídas uniformemente nos nós
 - chaves do nó original + nova chave
- Exemplo: Inserção de J



14

Inserção: Overflow Na Raiz

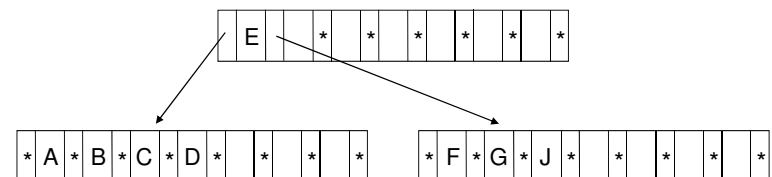
- **Passo 2** – Criação de uma Nova Raiz
 - existência de um nível mais alto na árvore permite a escolha da chave separadora
- Exemplo:



15

Inserção: Overflow Na Raiz

- **Passo 3** – Promoção de Chave
 - primeira chave do novo nó resultante do split é promovida para o nó raiz
- Exemplo:



16

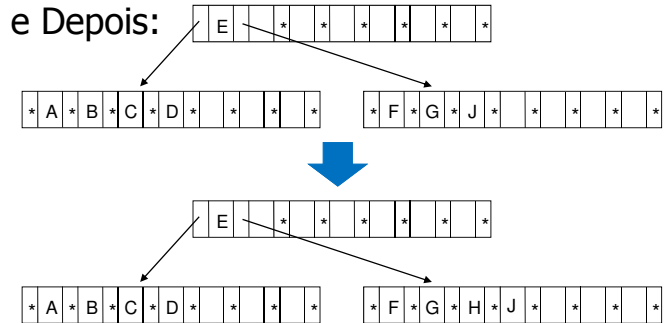
Inserção: Nós Folhas

- **Passo 1** – Pesquisa
 - árvore é percorrida até encontrar o nó folha no qual a nova chave será inserida
 - página correspondente é lida em RAM
- **Passo 2** – Inserção em Nó com Espaço
 - chave é inserida
 - página é reordenada e rescrita

17

Inserção: Nós Folhas

- Inserção em Nó com Espaço (Exemplo):
 - Inserção da chave H
- Antes e Depois:



18

Inserção: Nós Folhas

- **Passo 2** – Inserção em Nó Cheio (Overflow)
 - Particionamento (split):
 - criação de um novo nó folha
 - nó original → nó original + novo nó
 - distribuição uniforme das chaves nos dois nós
 - Promoção:
 - 1a chave do novo nó promovida a chave separadora no nó pai
 - reordenação e ajuste do nó pai para apontar para o novo nó
 - eventual propagação de overflow...

19

Exemplo

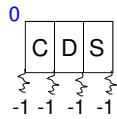
- Insira as seguintes chaves em um árvore B:
 - C S D T A M P I B W N G U R K E H O L J Y Q Z F X V
- Ordem da árvore B: 4
 - em cada nó (página de disco)
 - número de chaves: 3
 - número de ponteiros: 4

20

CSDTAM ...

- **1** – Inserção de C, S, D
 - criação do nó raiz

- C
- C S
- C D S

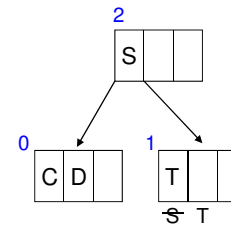


21

CSDTAM ...

- **2** – Inserção de T
 - nó raiz cheio

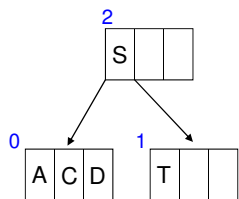
- split do nó
- criação de uma nova raiz
- promoção de S



22

CSDTAM ...

- **3** – Inserção de A
 - nó folha com espaço

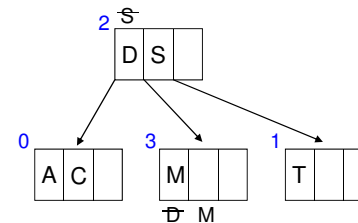


23

CSDTAM ...

- **4** – Inserção de M
 - nó folha 0 cheio

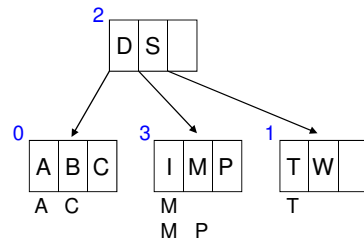
- split do nó
- promoção de D



24

... PIBWNGURK ...

- **5** – Inserção de P, I, B, W
 - nós folhas com espaço

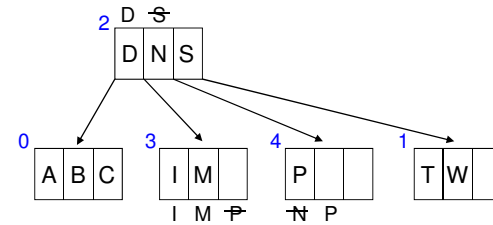


25

... PIBWNGURK ...

- **6** – Inserção de N
 - nó folha 3 cheio

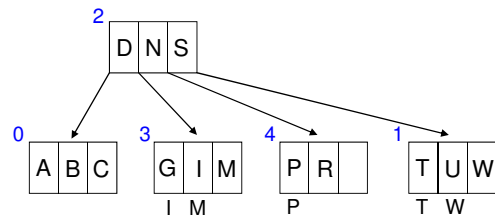
- split do nó
- promoção de N



26

... GURK ...

- **7** – Inserção de G, U, R
 - nós folhas com espaço

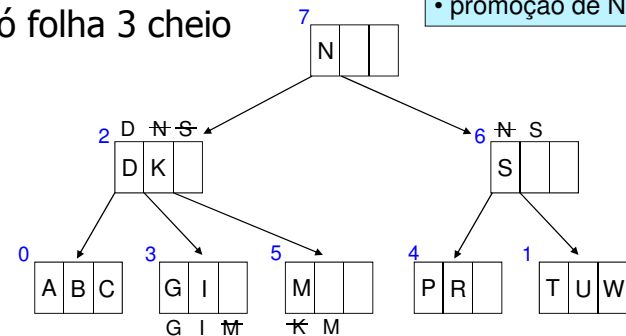


27

... GURK ...

- **8** – Inserção de K
 - nó folha 3 cheio

- split do nó 3
- promoção de K
- split do nó 2
- criação de uma nova raiz
- promoção de N



28



... E H O L J Y Q Z F X V

- Exercício:
 - Finalizar a construção da árvore...

29



Exercícios

- Na árvore B do exemplo anterior, insira a chave \$, sendo que $\$ < A$
- Insira as seguintes chaves em uma árvore B vazia:
 - C S D T A M P I B W N G U R K E H O L J Y Q Z F X V
 - diferentemente do exemplo anterior, escolha o último elemento do 1º nó para promoção após split por overflow
- Estude os pseudo-códigos para inserção de itens em árvores B na seção 9.9 de (Folk & Zoellick, 1987)
- **Outros Exercícios:** Capítulo 9 (Folk & Zoellick, 1987)

30



Bibliografia

- **M. J. Folk and B. Zoellick, *File Structures: A Conceptual Toolkit*, Addison Wesley, 1987.**

31