

Capítulo

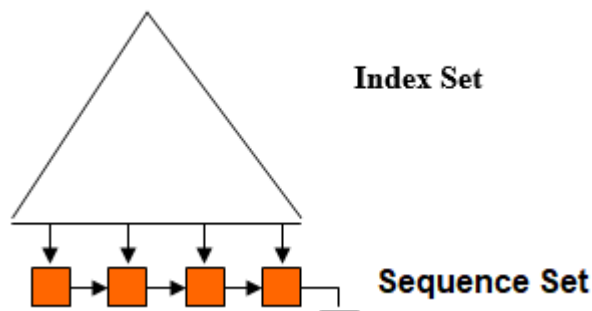
13

B+Tree

B+-tree

Introdução

A árvore B+-tree é uma extensão da B-tree. Na realidade é uma estrutura híbrida composta por uma B-tree e uma lista linear ordenada crescente encadeada. A B-tree está numa área denominada de **index set** utilizada para a pesquisa direta dos registros nos arquivos. A lista linear está numa área denominada de **sequence set** e é utilizada para pesquisa sequencial e dinâmica no arquivo.

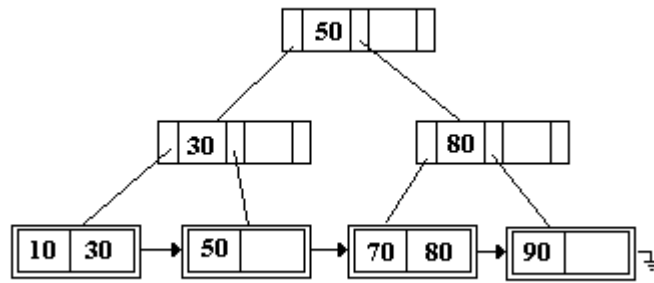


Index Set

É uma área estruturada na forma de B-tree e guarda as chaves que foram promovidas quando ocorreram um splitting. É utilizada para o acesso direto ao arquivo, isto é, quando necessitamos pesquisar uma determinada chave no arquivo devemos utilizar esta área para a pesquisa.

Sequence Set

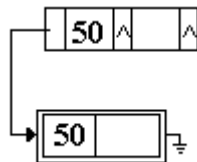
É uma área estruturada na forma de lista linear ordenada crescente encadeada e guarda todas as chaves que foram inseridas. É utilizada para o acesso sequencial ao arquivo ou para o acesso dinâmico, isto é, quando necessitamos pesquisar no arquivo chaves maiores ou iguais a uma determinada chave informada. Primeiramente visita-se o index set, e depois de achada a chave, faz-se uma varredura sequencial na lista para as chaves maiores.



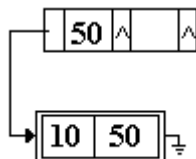
Construção Bottom-Up das B+-trees

Na construção das B+-tree a inserção é feita sempre nos nós folhas isto é no **sequence set**. Quando não existir mais espaço na célula pretendida faz-se um **splitting** e em seguida um **promotion** de forma similar à B-Tree.

Por exemplo, vamos construir uma B+-tree passo a passo. A célula da área de index set vamos considerar como de ordem 2 e célula da área de sequence set vamos considerar com duas chaves como no exemplo da figura anterior. Primeiramente vamos incluir a primeira chave 50. As duas áreas são inicializadas, uma célula para cada área, a chave é colocada na raiz do index set e como primeira chave no sequence set. A figura a seguir ilustra este processo.

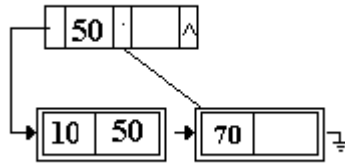


A seguir vamos incluir a chave 10. A pesquisa da folha a ser inserida é sempre feita pelo index set. Como a chave é menor que a primeira chave da raiz, deve-se procurar a folha do sequence set correspondente. Como já existe esta célula então a chave é incluída nesta célula. A figura a seguir ilustra este processo.

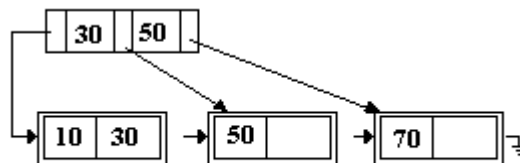


A próxima chave a incluir é a chave 70. A pesquisa da folha a ser inserida no index set nos leva a uma folha do sequence set que não existe. Como a chave é maior

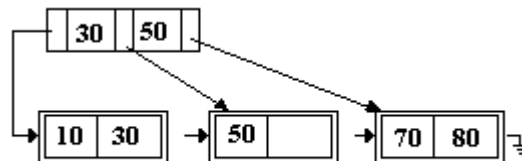
que a primeira chave da raiz, deve-se procurar a folha do sequence set correspondente mas não existe folha uma vez que o ponteiro está aterrado. Neste caso, cria-se uma nova célula no sequence set e a chave é incluída nesta célula. A figura a seguir ilustra este processo.



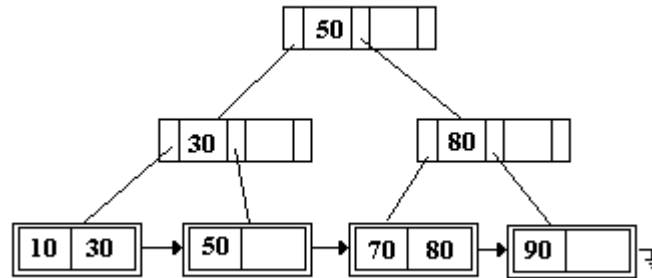
A seguir vamos incluir a chave 30. A pesquisa da folha a ser inserida no index set nos leva à primeira célula do sequence set. Pode-se observar que não existe espaço para a inserção na célula pretendida. Ocorre um splitting da célula da área do sequence set e as chaves são redistribuídas e com uma diferença para as inserções nas B-Tree, a chave que é promovida também fica na área do sequence set. A figura a seguir ilustra este processo.



A próxima chave a incluir é a chave 80. A pesquisa da folha a ser inserida no index set nos leva à terceira célula do sequence set. Pode-se observar que existe espaço para a inserção na célula pretendida. A inclusão é feita de forma normal. A figura a seguir ilustra este processo.



A seguir vamos incluir a chave 90. A pesquisa da folha a ser inserida no index set nos leva à terceira célula do sequence set na figura anterior. Pode-se observar que não existe espaço para a inserção na célula pretendida. Ocorre um splitting da célula da área do sequence set e as chaves são redistribuídas e com uma diferença para as inserções nas B-Tree, a chave que é promovida (80) também fica na área do sequence set. Quando a chave 80 é promovida, na B-Tree, não existe espaço para a inserção. Ocorre um splitting da célula



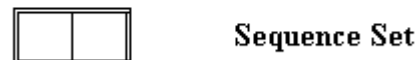
da área do index set e as chaves são redistribuídas de forma similar nas B-Tree. A figura a seguir ilustra este processo.

Exercícios

1. Construir uma B+-Tree com as chaves 10, 20, 40, 30, 70, 90, 81, 33, 45, 65, 78, 99, 11 e 15. Considere as células da área do index set e do sequence set representadas abaixo.



2. Construir uma B+-Tree com as chaves PSC, INF, QUI, SET, ENG, MED, FIS, ARQ, MAT, GEO, HIS, EDU. Considere as células da área do index set e do sequence set representadas abaixo.

***Bibliografia***

1. HOROWITZ, E. ; SAHNI, Sartaj. Fundamental of Data Structures in Pascal. New York, Computer Science Press, 1994.
2. ZIVIANI, N. *Projeto de Algoritmos com Implementação em Pascal e C*. 4ª ed. São Paulo: Pioneira, 1999.
3. Folk, Michael J. & Zoelik, Bill, *File Structures*, Addison-Wesley , 1992 (2ª Edition);
4. Tharp, Alan j., *File organization and Processing*, John Wiley & Sons, Inc., 1988;
5. AHO, A. V. ; HOPCROFT, John E. Data Structures and Algorithms. Massachusetts: Addison Wesley, 1983;