CSE 674 Advanced Data Structures

Multiway Trees

Andrew C. Lee

EECS Dept, Syracuse U.

Contents

We will discuss

1. Multiway Search Trees
2. B Trees and its variants

Binary Search Trees

Binary Search Trees

1. When volume of data is large, it may not all fit in the main memory
2. Issues: page faults (disk access cost more time)
3. Possible remedy: use multiway trees

Multiway Trees

1. Each non-leaf node test m keys
2. Has $m + 1$ branches
3. TRY to adjust the number of keys dynamically to maintain balance

This leads to the development of a data structure called the B trees

B Trees

1. Node structure

$$[p_0 k_1 p_1 k_2 \dots k_{j-2} p_{j-2} k_{j-1} p_{j-1} k_j p_j]$$

2. Definition
   - All leaf nodes appear at the same level and contain no key values
   - Unless the root is a leaf node, it has at least 2 sons
   - All nodes other than the root and the leaf nodes have at least $\lceil \frac{m}{2} \rceil$ children
   - A node has a maximum of m children
   - A non-leaf node with j key values has $j + 1$ sons

Special Case: 2-3 Trees

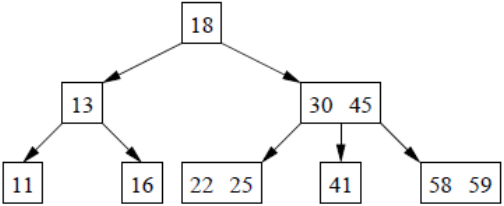

Figure: B-Tree of Order 3: 2-3 trees

Major Operations

1. Search
2. Insert
3. Delete

We will make use of additional illustrations to show how the operations can be implemented