Multiway Trees

Tree

• A (free) tree is any set of points (called vertices) and any set of pairs of distinct vertices (called edges or branches) such that (1) there is a sequence of edges (a path) from any vertex to any other, and (2) there are no circuits, that is, no paths starting from a vertex and returning to the same vertex.
• A rooted tree is a tree in which one vertex, called the root, is distinguished.
• An ordered tree is a rooted tree in which the children of each vertex are assigned an order.
• A forest is a set of trees. We usually assume that all trees in a forest are rooted.
• An orchard (also called an ordered forest) is an ordered set of ordered trees.
Examples

- Free trees with 4 or fewer vertices (Arrangement of vertices is irrelevant)
- Rooted trees with 4 or fewer vertices
- Ordered trees with 4 or fewer vertices

Implementation

- First child black
- Next child color
Tries
B-Trees

- Idea: In external search disk access is expensive. How can we store more than 2-way information in a block?

- Solution: construct multiway search tree.
B-Trees

- A B-tree of order $m$ is an $m$-way search tree in which
  - All leaves are on the same level.
  - All internal nodes except the root have at most $m$ non-empty children, and at least ceiling($m/2$) nonempty children.
  - The number of keys in each internal node is one less than the number of its children, and these keys partition the keys in the children in the fashion of a search tree.
  - The root has at most $m$ children, or as few as 2 if it is not a leaf, or none if the tree consists of the root alone.

Example (m=4)
Insertion

- B-trees grow at the root, not the leaves.
- Find the leaf where the new key belongs and insert it.
- If the leaf now has too many keys, split it into two nodes on the same level, but do not put the median key into either new node.
- Move up one level, insert the median key in this parent node, and repeat the splitting process if necessary.
- If the root node splits, then the resulting median key goes into a new root and the tree grows in height.
Insertion Example

Deletion
Deletion Example

1. Delete h, r:

   ![Tree Diagram]

   - Promote s and delete from leaf.

2. Delete p:

   ![Tree Diagram]

   - Pull s down, pull r up.

Deletion Example (cont..)

3. Delete d:

   ![Tree Diagram]

   - Combine:

Deletion Example (cont..)

   ![Tree Diagram]

   - Combine: