15CSE374
INTRODUCTION TO DATA STRUCTURES AND ALGORITHMS

Sarath tv

Last Lecture

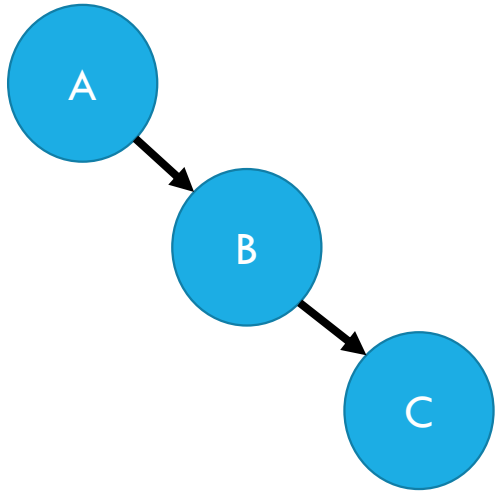
- Unbalanced Tree.
- AVL Tree.
- Height Invariant.
- Balance factor.

Rotations

- Four kind of rotations
  - Left rotation
  - Right rotation
  - Left-Right rotation
  - Right-Left rotation

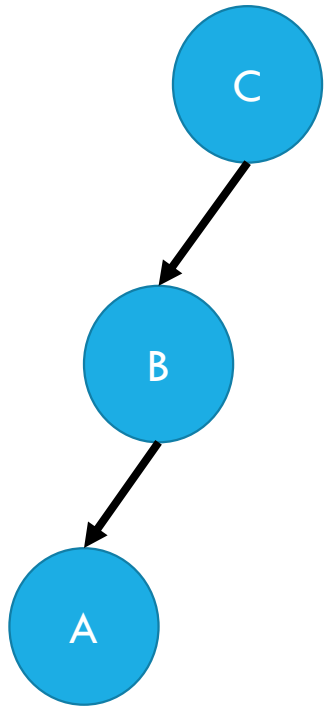
Left Rotation

- If a tree becomes unbalanced because of a node getting inserted into the right subtree of the right subtree.



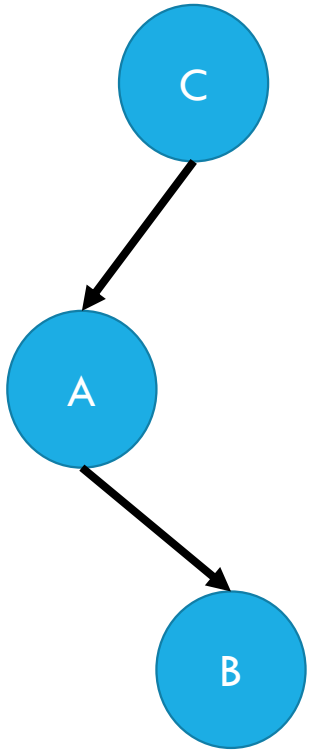
Right Rotation

- If a node is inserted in the left subtree of the left subtree ,which causes tree to be unbalanced.



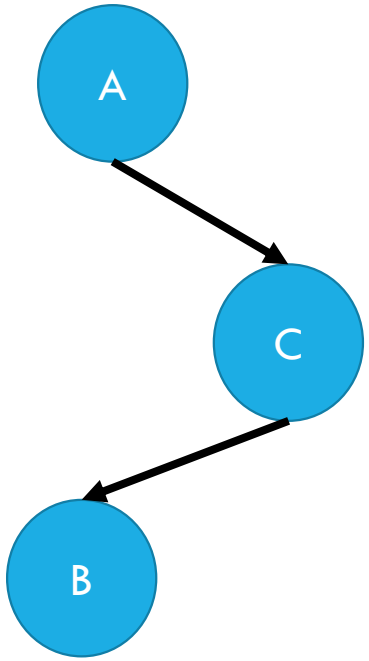
Left Right Rotation

- A node inserted into right subtree of the left subtree.
- Combination of left rotation followed by right rotation.



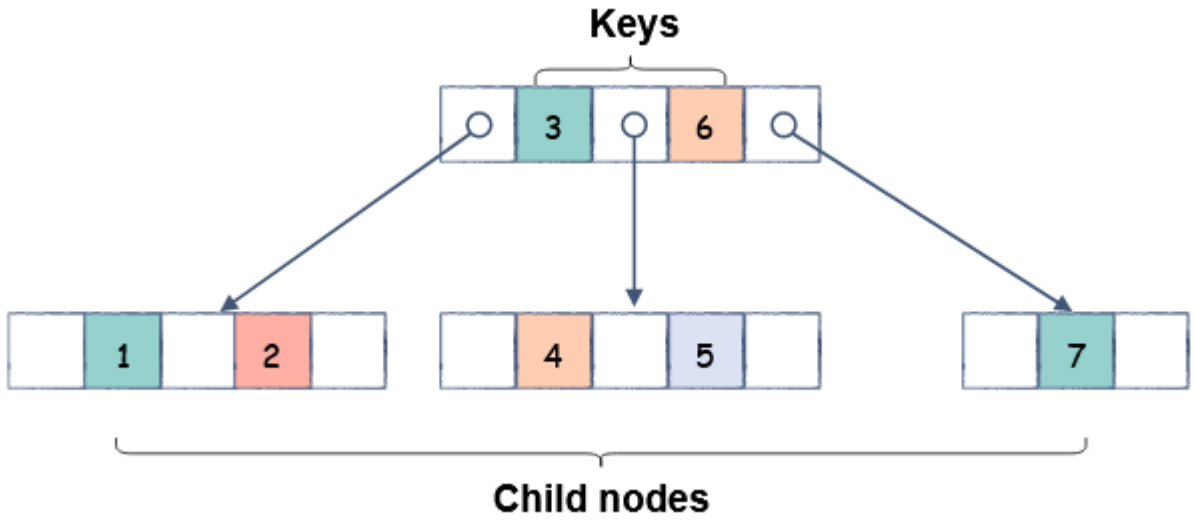
Right Left Rotation

- A node inserted into left subtree of the right subtree.
- Combination of right rotation followed by left rotation.



B-Tree

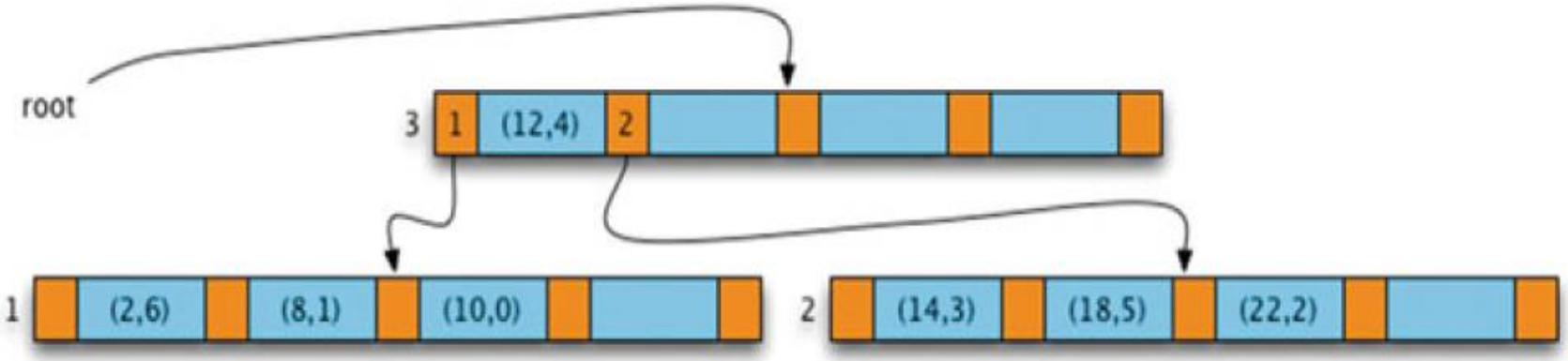
- Self balancing tree data structure.
- O(log n)
- Each node in a B-tree of order m can have at most m children and m-1 key.
- Stored data is sorted similar to a BST , but B-Tree can have more than two child nodes.



- B-Tree can have more than one key.
- Height of tree relatively small.
- Main usage in relational databases to implement an operation called “Join”.
- RDB consists of entities and relationships between those entities.

B Tree Organization

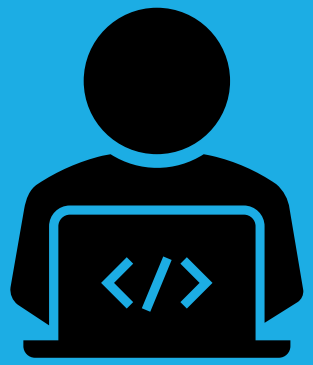
- Each node in a B-Tree contains pointers to other nodes and items in an alternating sequence.
- The items in a node are arranged sequentially in order of their keys.
- A pointer to the left of an item points to another B-Tree node that contains items that are all less than the item to the right of the pointer.
- A pointer to the right of an item points to a node where all the items are greater than the item.



- There will always be one more pointer than items in a node.
- Degree of a B-Tree is the minimum number of items that a B-Tree node may contain, except for the root node.
- Capacity of the node is always twice its degree.
- For our example degree is 2 and capacity is 4.

The requirements of a B-Tree are

- Every node except the root node must contain between *degree* and *2*degree* items.
- Every node contains one more pointer than the number of items in the node.
- All leaf nodes are at the same level within a B-Tree.
- The items within a B-Tree node are ordered in ascending (or descending) order.
- All nodes have their items in the same order, either ascending or descending.
- The items in the subtree to the left of an item are all less than that item.
- The items in the subtree to the right of an item are all greater than that item.



THANK YOU!!!!!