4/26/23

ADS Theory Assignment

Krishnaraj PT
PA - 20, A1

Q.1. Explain AA trees with example.

→ We must first learn what are Red Black binary trees:

(A) Red - Black Tree

→ Special type of binary tree in which each node is either Red or black. It is a self balancing Binary Search Tree.

(A) Properties:

1. Root is black
2. Leaf is always black
3. Children of a red node are black
4. All leaves have same black depth
5. There are no 2 adjacent red nodes

(A) Why?

This greatly reduces time required for insertion and deletion; and makes it better than AVL trees. Insertion, Deletion and searching always at most takes O(log n) time.

(★) How ?

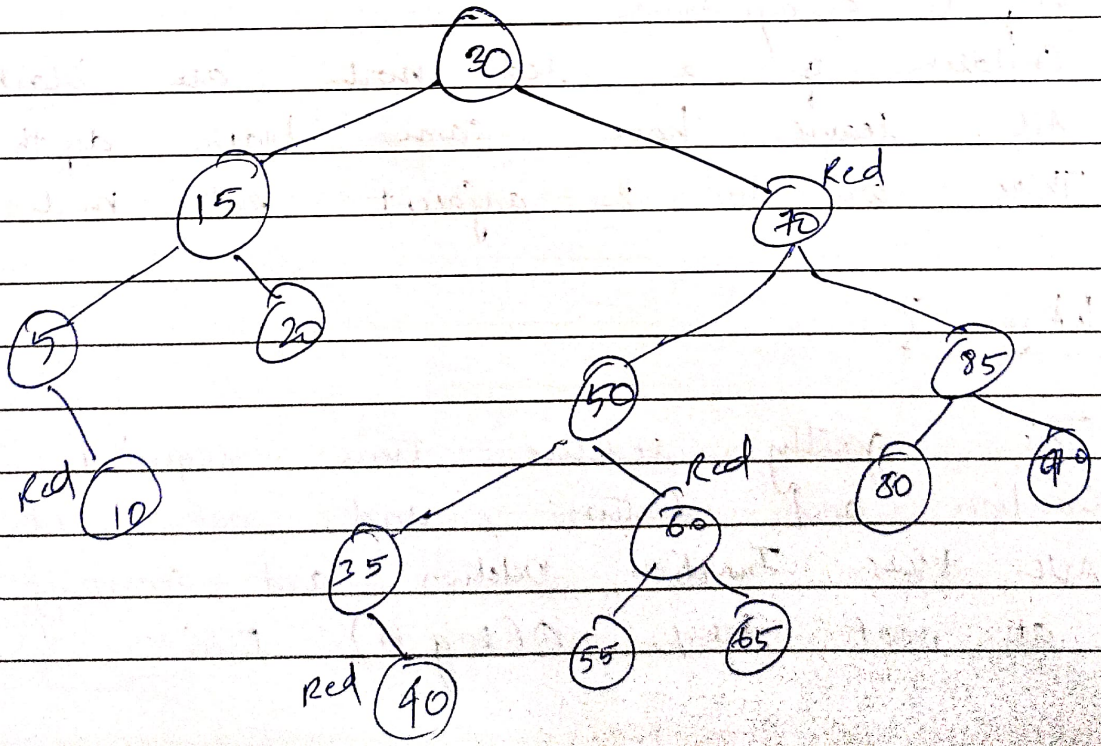
→ These trees ensure balance by making sure a chain of 3 nodes is not possible in the Red - Black tree.

(★) A-A trees

→ They are a variation of Red black trees and a form of binary trees.

→ They use the concept of levels in balancing binary trees. The level is used for balancing information.

→ They follow the same rules as red black trees with the addition of a new rule that red nodes cannot be present as left children



Q. 2. Explain B+ tree

→ B+ tree is an extension of B tree which allows efficient insertion, deletion and search operation.

→ In B tree, keys and records can both be stored in the internal and leaf nodes, whereas, in B+ tree, records can only be stored on the leaf and the internal nodes can only stored the key values.

→ The leaf nodes of a B+ tree are linked together in the form of a singly linked list to make the search more efficient.

→ B+ trees are used to store the large amounts of data which cannot be stored in the main memory. Due to the fact the size of the main memory ~~Due to the fact~~ is always limited, the internal nodes of B+ tree are stored in the main memory whereas leaf nodes are stored in the secondary memory

(*) The advantages of B+ tree are:

→ Records can be fast fetched in equal number of disk access.

→ The ~~advantages of~~ B+

→ Heights of the tree remains balanced and less compared to B tree.

→ We can access the data stored in a B+ tree sequentially as well as directly.

→ Keys are used for indexing.

→ Faster search queries as the data is stored only on the leaf nodes.

eg. B+ tree of order 3.