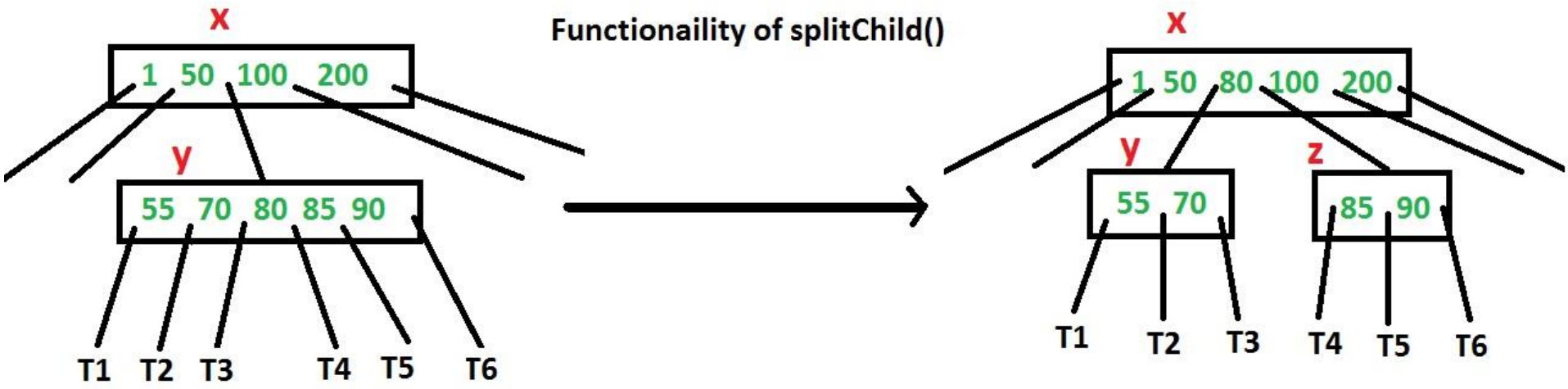
Other Balanced Trees

EECS 233

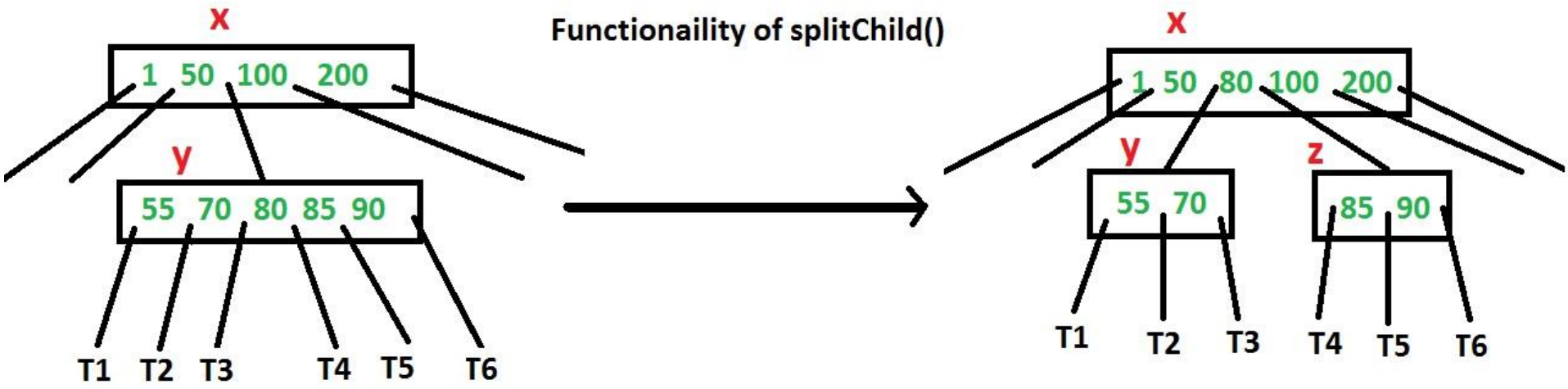
Insertion into B-Trees

- new key is always inserted at leaf node
- before inserting a key to node, we make sure that the node has extra space
- use splitChild() to split a child of a node



Insertion into B-Trees

- On the way to the insertion point, split each full node
- The advantage of splitting before is, we never traverse a node twice



Insertion Example

- tree of minimum degree t=3
- insert integers 10, 20, 30, 40, 50, 60, 70, 80 and 90

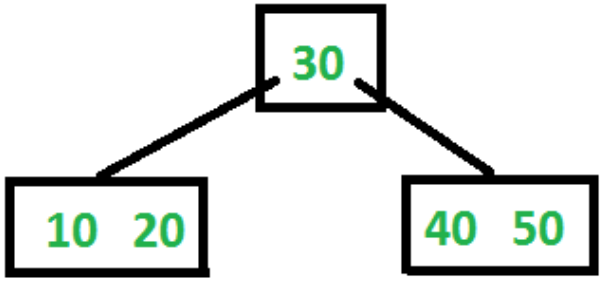
Insert 10

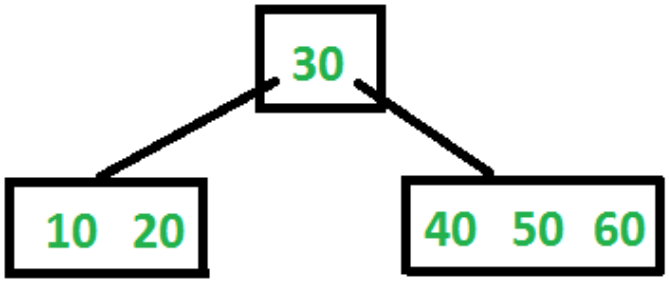
Insert 20, 30, 40 and 50

Insert 60

Insert 20, 30, 40 and 50

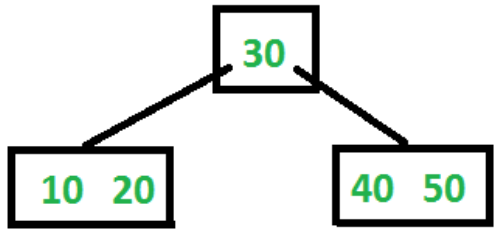
Insert 60

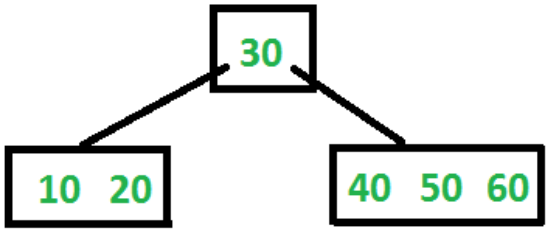




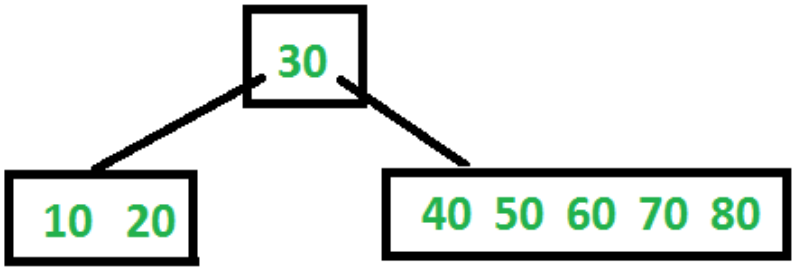
Insert 70 and 80

Insert 60



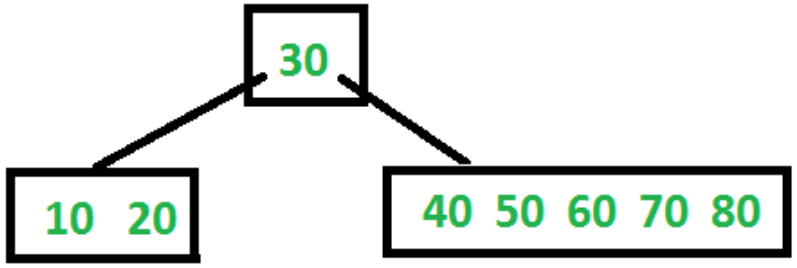


Insert 70 and 80

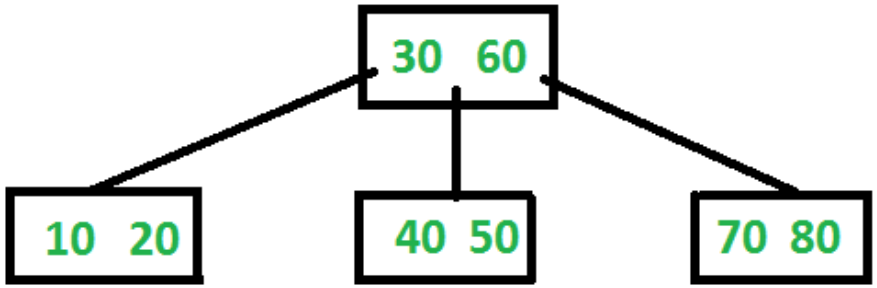


Insert 90

Insert 70 and 80



Insert 90

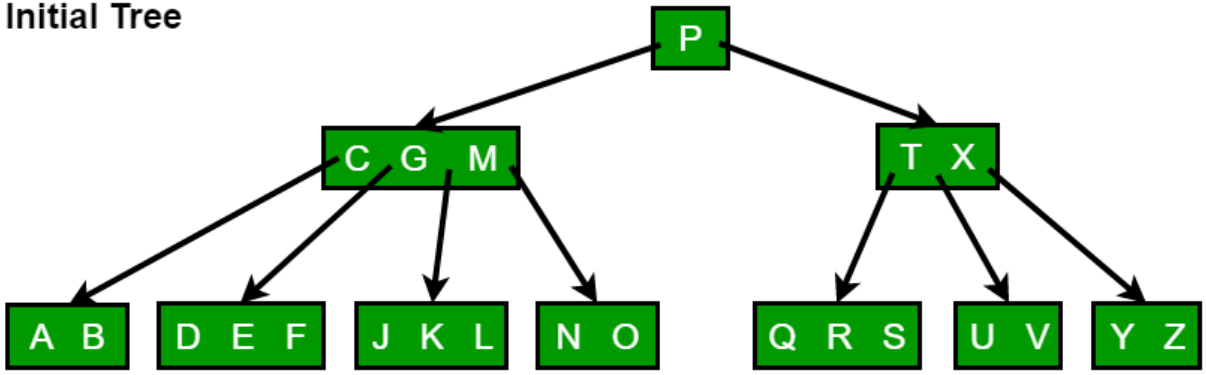


Deletion from B-trees

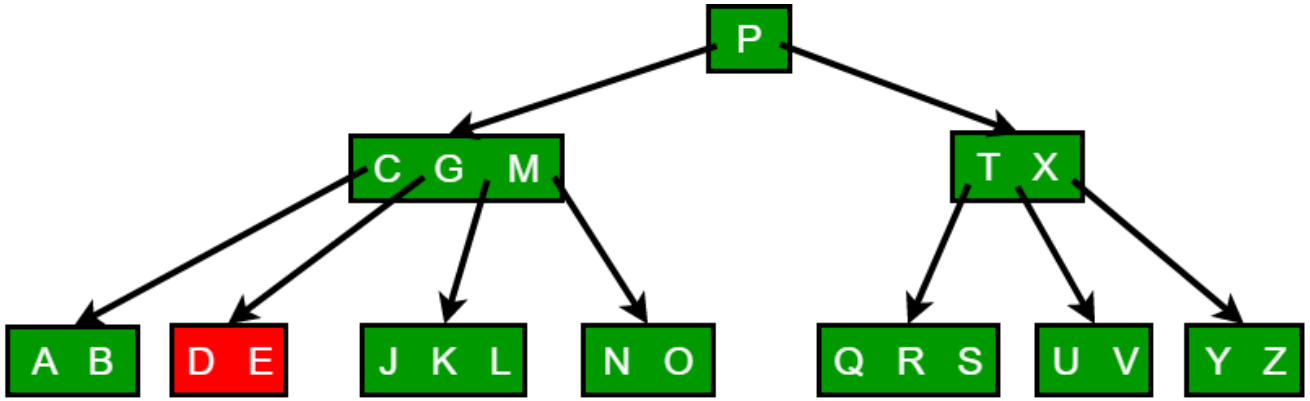
- Ensure that a node doesn't get too small during deletion
- Back up if a node (other than the root) along the path to where the key is to be deleted has the minimum number of keys

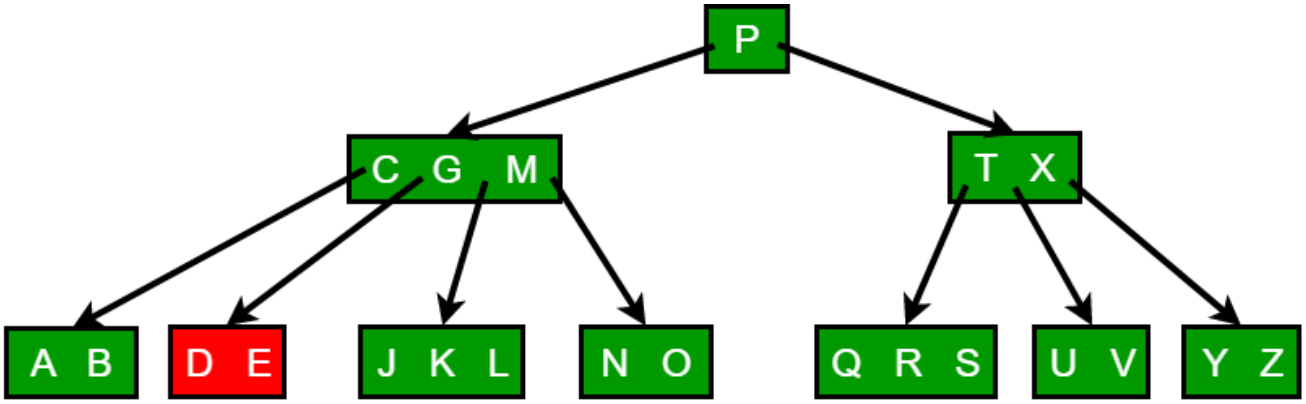
t=3

(a) Initial Tree

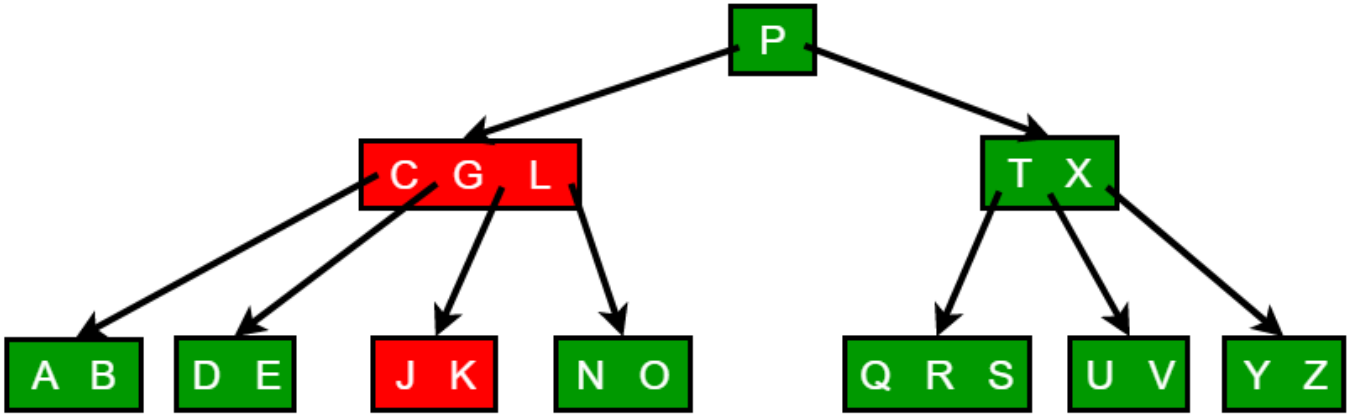


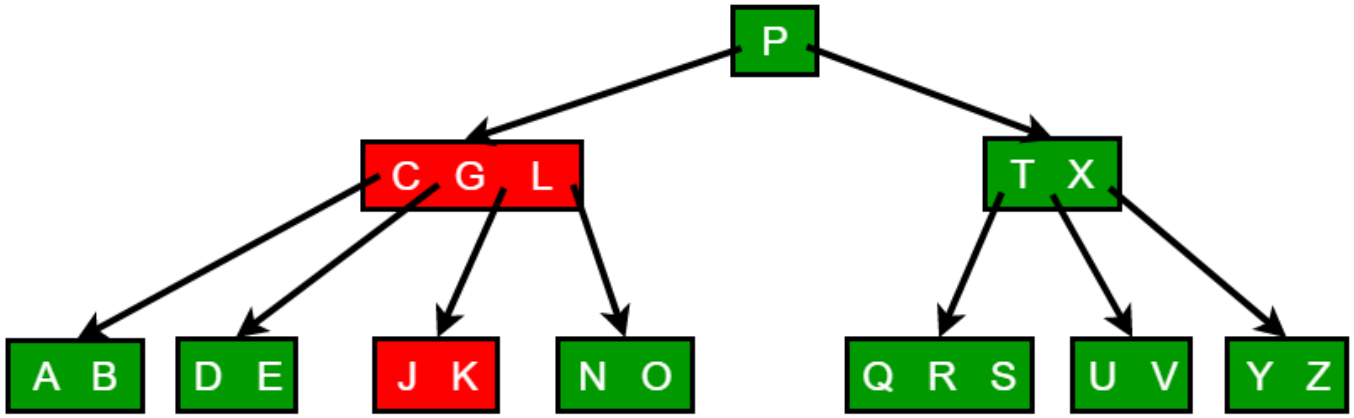
Delete F



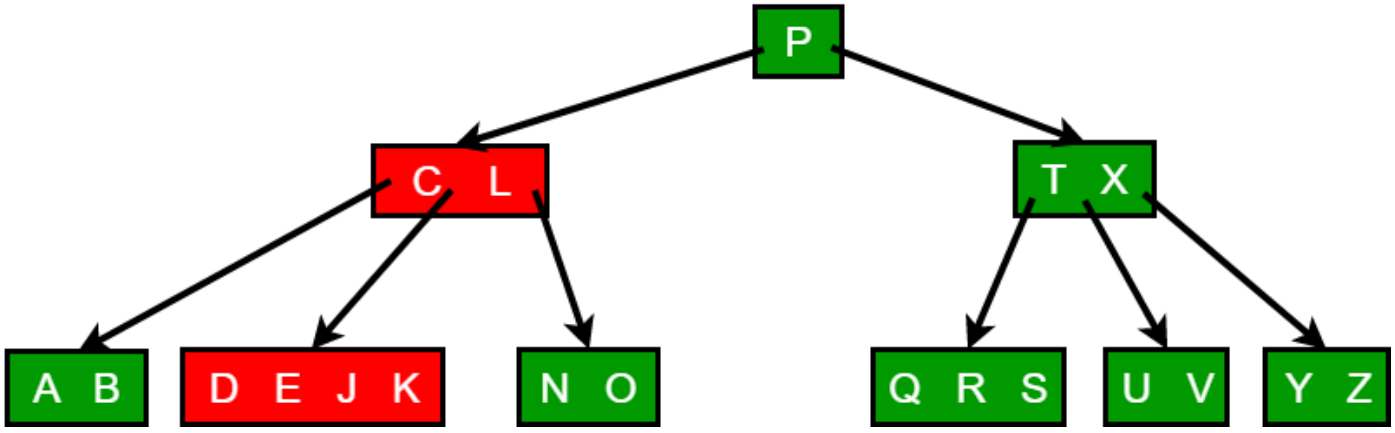


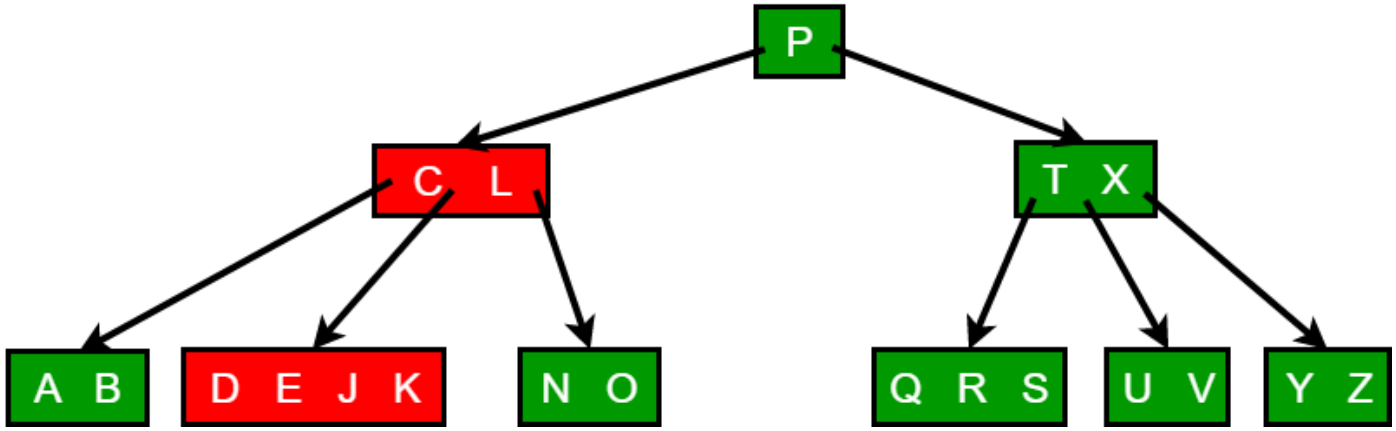
Delete M



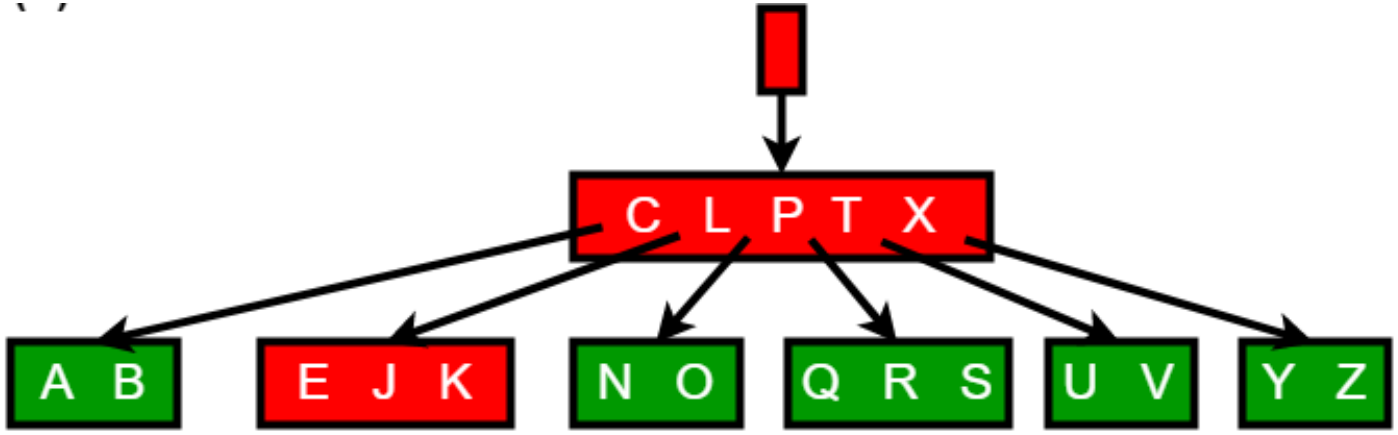


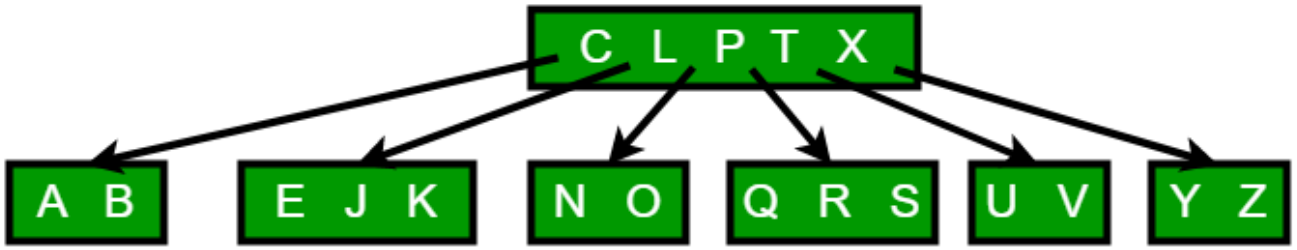
Delete G





Delete D





Delete B