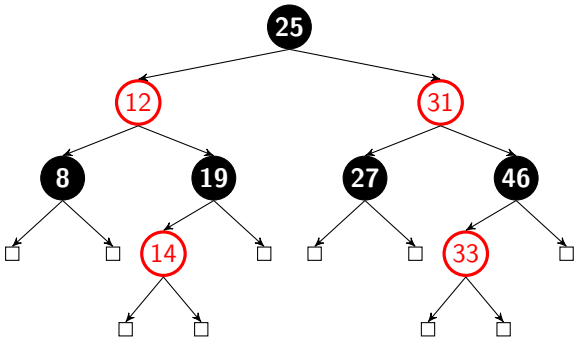
CS 61B Spring 2022

Graphs & LLRBs

Discussion 13: April 18, 2022

1 Balanced Search Trees

(a) Convert the red-black tree into a 2-4 tree. Solid nodes are black.

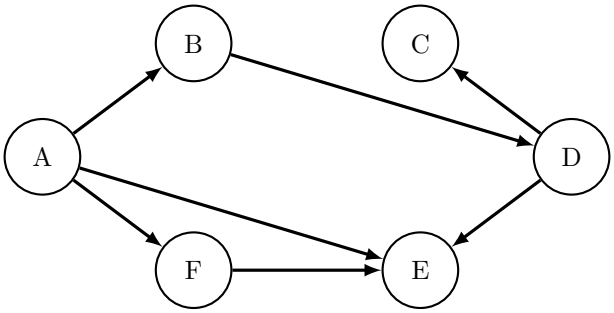


(b) Insert the keys 13 and 17 into the resulting 2-4 tree. Assume that, if a node has 4 keys, we choose to push up the left of the 2 middle keys (so the 2$^{\text{nd}}$ key from the left).

(c) Convert the resulting 2-4 tree into a valid left-leaning red-black tree.

(d) Given a 2-4 tree containing N keys, describe how you can obtain the keys in sorted order in worst case $O(N)$ time.

(e) If a 2-4 tree has depth H (that is, the leaves are at a distance of H from the root), what is the maximum number of comparisons done in the corresponding red-black tree to find whether a certain key is present in the tree?



2 Graph Representation

Represent the graph above with an adjacency list and an adjacency matrix representation.

3 Searches and Traversals

Run depth first search (DFS) preorder, DFS postorder, and breadth first search (BFS) on the graph above, starting from node A. List the order in which each node is first visited. Whenever there is a choice of which node to visit next, visit nodes in alphabetical order.

4 Topological Sorting

Give a valid topological ordering of the graph. Is it unique?