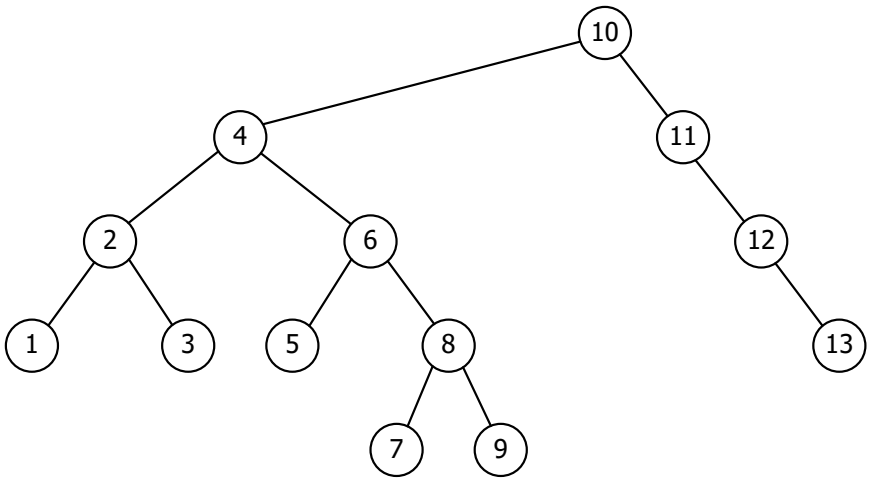
Data Structures and Algorithms

Assignment 4

1. Write a program to implement a basic binary search tree and operations.
   a) Implement all routines introduced in class, e.g., MakeEmpty(), Find(), FindMax(), Insert(), Delete(), Retrieve(), InOrder(), PreOrder(), PostOrder(), LevelOrder().
   b) Based on the routines written in a), write a function, e.g., named as ConstructTree(), to construct the tree T as follows, by inserting the elements one by one with Insert() in the function (Please initialize the sequence of elements in an array).
   c) In main(), call the ConstructTree() to construct the tree, and then call InOrder(), PreOrder(), PostOrder() and LevelOrder() to output the In-Order, Pre-Order, Post-Order and Level-Order traverse sequences to screen.



2. Write a program to implement a basic AVL and operations.
   a) Implement routines introduced in class, especially SingleRotateWithRight() and DoubleRotateWithRight().
   b) Implement a routine, as named as PrintTreeEdge(), to output a tree to screen by printing all edges in the tree, in form of parent -> node, like 5 ->7, 9->6, etc. (Hint: basically, it is a traverse operation).
   c) Store a sequence of elements separated by space to construct an AVL tree in a plain text file, e.g., TreeData.txt = "3 2 1 4 5 6 7 10 11 12 13 14 15 16 8 9".
   d) In main(), read data from the text file till the end of the sequence in the text file, and call Insert() (surely, together with necessary rotate operations) to construct the corresponding AVL tree one element by one element, then output the finally derived AVL tree to screen by calling PrintTreeEdge().

Note:

- Due date: Dec. 5th.