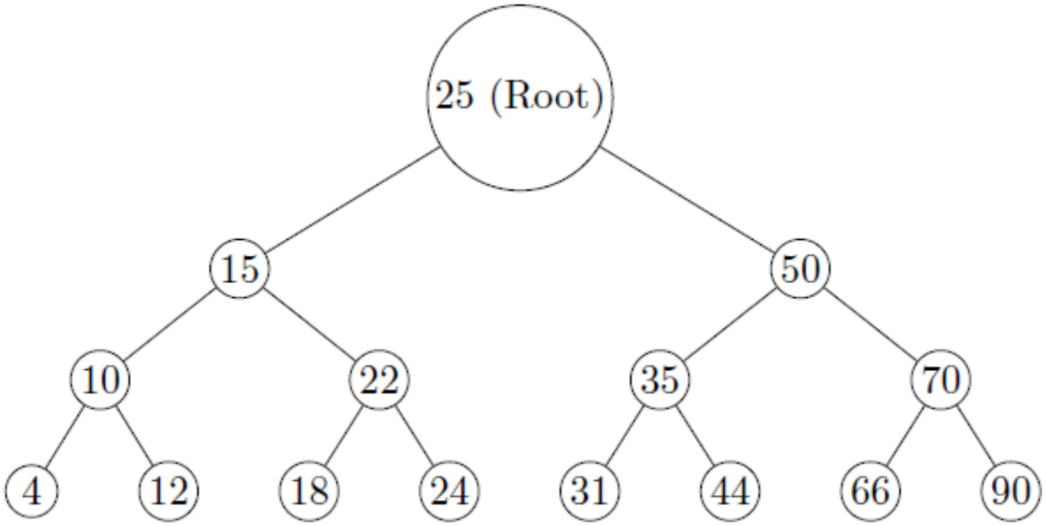
CS401 Lab 9

This lab is to be completed individually.

This lab is for you to understand Tree data structure implementation.

What to do?

Create a binary search tree (BST). You can use the below example to build the tree:



Perform and demonstrate the following specifications using Java:

1. Find your binary search tree's maximum depth
The maximum depth is the number of nodes along the longest path from the root node down to the farthest leaf node
EX: Given the tree above, we have maximum depth = 4

2. Calculate size of the tree with recursive and iterative methods

3. Print traversal of tree with pre-order, post-order and in-order methods
If using the above tree, make sure you have the following outputs to ensure the correctness of your program:

- In-order (Root) visits nodes in the following order:
  4, 10, 12, 15, 18, 22, 24, 25, 31, 35, 44, 50, 66, 70, 90

- Pre-order traversal visits nodes in the following order: 25, 15, 10, 4, 12, 22, 18, 24, 50, 35, 31, 44, 70, 66, 90

- Post-order traversal visits nodes in the following order: 4, 12, 10, 18, 24, 22, 15, 31, 44, 35, 66, 90, 70, 50, 25

Make sure that your code is well documented i.e., in-line comments with a simple README would be ideal. For instance, every function and complex portion of code should have comments that describe what it does.

What to turn in?

1. Source code

2. Your program's outputs in a PDF file

3. JAR file.

4. README file to demonstrate how your program works. Include a command to determine how to run the JAR file.

Please submit on Blackboard before the assigned due date