# Lab 6 : Binary Search Tree

# Week beginning 2nd November, 2015

The code for BST is given on shared drive.

1. Use debug mode to step through the insertSub method to see the recursive calls.

2. Write a recursive contains method.

3. Write an iterative contains method. Which version – iterative or recursive - is more efficient?

4. Write an iterative version of insert method i.e. using a loop. Which version is more efficient?

5. Use debug mode to step through the printSub method to see the recursive calls.

6. Change the printSub method to perform

* preorder traversal
* postorder traversal

Check that the output from these methods is what you expect.

7. Write a non-recursive version of preorder traversal. What data structure is required?

Use the following algorithm:

Push the root (reference to root node) onto the stack.

While the stack is not empty

pop the stack and visit it

push its two children (reference to children) //what order do you push children??

To use a stack, use Deque interface as Java recommends. Use LinkedList as implementation of Deque.