TCP1201 Object-Oriented Programming and Data Structures

Lab13-14 Recursion and Binary Search Trees (BSTs)

Exercise 1: Sum of Digits Recursion

1) Write an **iterative (non-recursive)** method to compute the sum of the digits in an integer. Use the following method header:

```
public static int sumDigitsIterative(long n)
```

For example, sumDigits(234) returns 2 + 3 + 4 = 9.

2) Write a **test program** that prompts the user to enter an integer and displays its sum.

3) Write a **recursive** method that performs the same sum of digits. Test your recursive method. Use the following method header:

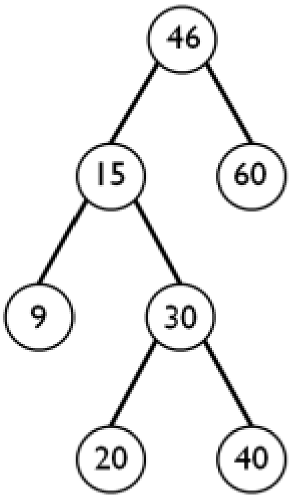
```
public static int sumDigitsRecursive(long n)
```

Sample run:

```
Enter a positive long integer: 123456
(Iterative) The sum of digits in 123456 is 21
(Recursive) The sum of digits in 123456 is 21
```

Exercise 2: Basics of Binary Search Trees

Consider the following **binary search tree**:



a) What node(s) are:
   i. Root?
   ii. Leaves?
   iii. Parents of 30?
   iv. Children of the node 15?
   v. Search path for the following targets:

i. 60
ii. 30
iii. 72
iv. 12

b) State the results of **preorder**, **inorder**, and **postorder traversals** on the above BST.

```
Preorder :
Inorder  :
Postorder:
```

c) Which traversal algorithm traverses all nodes in a BST in **ascending order**?

d) Draw a BST after inserting nodes 90, 25, 45, 27, 57, and 10 sequentially into the above BST.

Exercise 3: Implementing BST

Without copy and paste from the code in lecture, define a **BST** class that supports the operations listed in the sample run below. Write a test program to test your BST class. Your test program shall produce the following output.

Sample run:

```
Inorder  :
Preorder :
Postorder:
Size: 0
1. Insert specified integer
2. Search specified integer
3. Clear
0. Exit
Command > 1 45
true

Inorder  : 45
Preorder : 45
Postorder: 45
Size: 1
1. Insert specified integer
2. Search specified integer
3. Clear
0. Exit
Command > 1 45
false

Inorder  : 45
Preorder : 45
Postorder: 45
Size: 1
1. Insert specified integer
2. Search specified integer
3. Clear
```

```
0. Exit
Command > 1 61
true

Inorder  : 45 61
Preorder : 45 61
Postorder: 61 45
Size: 2
1. Insert specified integer
2. Search specified integer
3. Clear
0. Exit
Command > 1 26
true

Inorder  : 26 45 61
Preorder : 45 26 61
Postorder: 26 61 45
Size: 3
1. Insert specified integer
2. Search specified integer
3. Clear
0. Exit
Command > 1 98
true

Inorder  : 26 45 61 98
Preorder : 45 26 61 98
Postorder: 26 98 61 45
Size: 4
1. Insert specified integer
2. Search specified integer
3. Clear
0. Exit
Command > 2 11
false

Inorder  : 26 45 61 98
Preorder : 45 26 61 98
Postorder: 26 98 61 45
Size: 4
1. Insert specified integer
2. Search specified integer
3. Clear
0. Exit
Command > 2 61
true

Inorder  : 26 45 61 98
Preorder : 45 26 61 98
Postorder: 26 98 61 45
1. Insert specified integer
2. Search specified integer
3. Clear
```

```
0. Exit
Command > 3

Inorder  :
Preorder :
Postorder:
Size: 0
1. Insert specified integer
2. Search specified integer
3. Clear
0. Exit
Command > 0
```