LAB 11: Binary Search Trees

Statement Purpose:

Purpose of this Lab is to familiarize the students with the use of Binary Search Trees data structure in writing simple Java programs. Another aim is to teach the students how to implement Binary Search Tree data structure using linked list. The students are given small tasks related to BST which they complete during the lab session under the supervision of the lab instructor. This helps them understand the concepts well which they learn in their lectures.

Activity Outcomes:

The students will learn how to write methods related to some actions performed on Binary Search Trees. They will understand how to write simple methods to

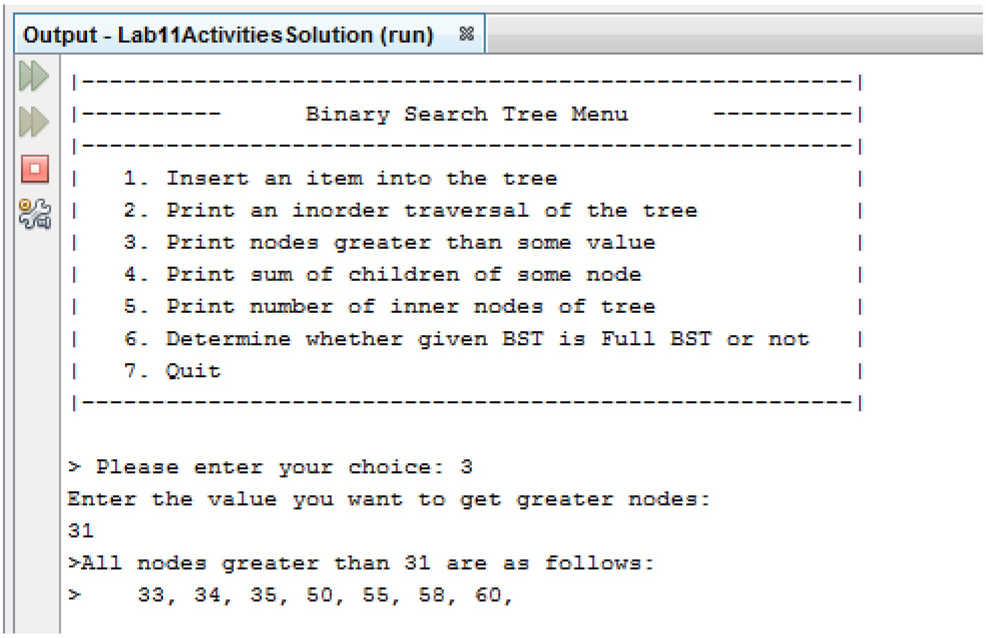
- print all nodes of BST which are greater than some specific value
- find the sum of children of some BST node
- count number of interior nodes of BST
- determine whether the given BST is a Full BST or not

LAB EXERCISES: (80 Minutes)

1. Write a method GreaterThan() in intBST class which prints all nodes of the BST which are greater than some specific value.

   Sample Run:

   If we insert the nodes 50, 30, 25, 35, 33, 34, 60, 55 and 58and choose the option 3, it will display the following output:



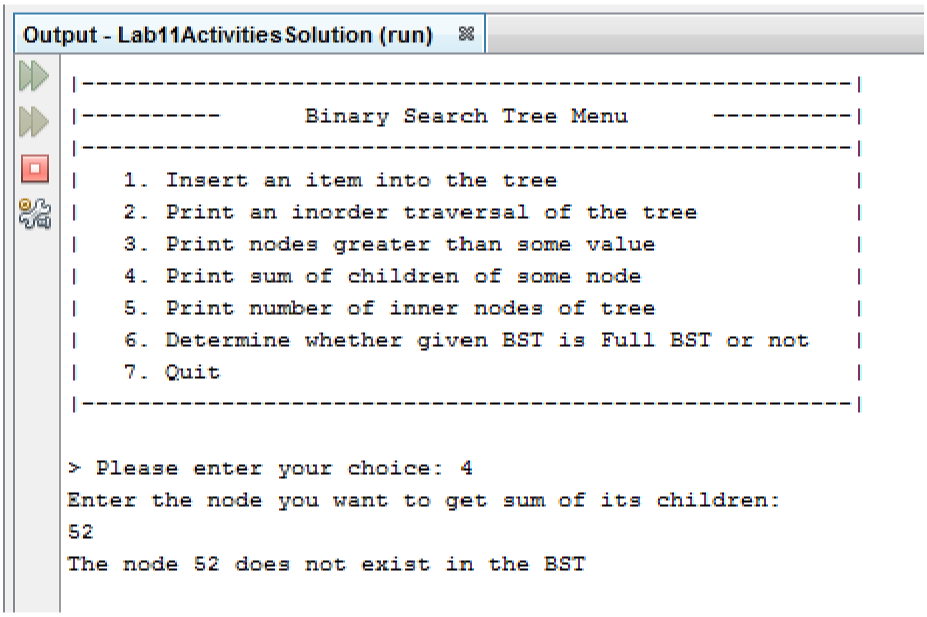
```
|-------------------------------------------------------|
|----------      Binary Search Tree Menu      ----------|
|-------------------------------------------------------|
|   1. Insert an item into the tree                     |
|   2. Print an inorder traversal of the tree           |
|   3. Print nodes greater than some value              |
|   4. Print sum of children of some node               |
|   5. Print number of inner nodes of tree              |
|   6. Determine whether given BST is Full BST or not   |
|   7. Quit                                             |
|-------------------------------------------------------|

> Please enter your choice: 3
Enter the value you want to get greater nodes:
31
>All nodes greater than 31 are as follows:
>    33, 34, 35, 50, 55, 58, 60,
```

2. Write a method SumChildren() in intBST class which finds and returns sum of children of some BST node.If this node does not have any children, program should display the message: “The node n does not have any children”.

Sample Run 1:

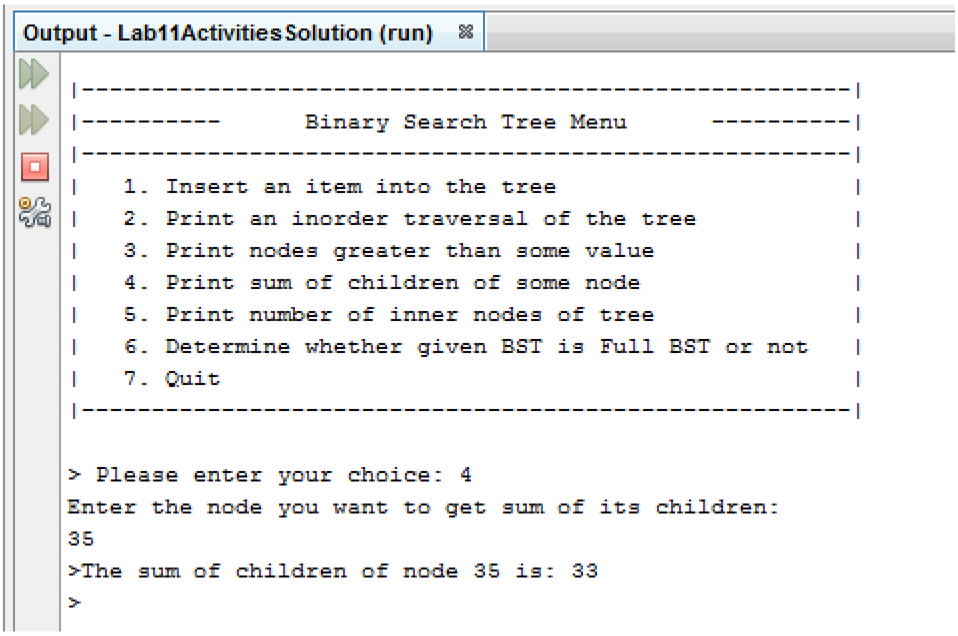
If we insert the nodes 50, 30, 25, 35, 33, 34, 60, 55 and 58 and choose the option 4, it will display the following output:



```
|-------------------------------------------------------|
|----------      Binary Search Tree Menu      ----------|
|-------------------------------------------------------|
|    1. Insert an item into the tree                    |
|    2. Print an inorder traversal of the tree          |
|    3. Print nodes greater than some value             |
|    4. Print sum of children of some node              |
|    5. Print number of inner nodes of tree             |
|    6. Determine whether given BST is Full BST or not  |
|    7. Quit                                            |
|-------------------------------------------------------|

> Please enter your choice: 4
Enter the node you want to get sum of its children:
52
The node 52 does not exist in the BST
```

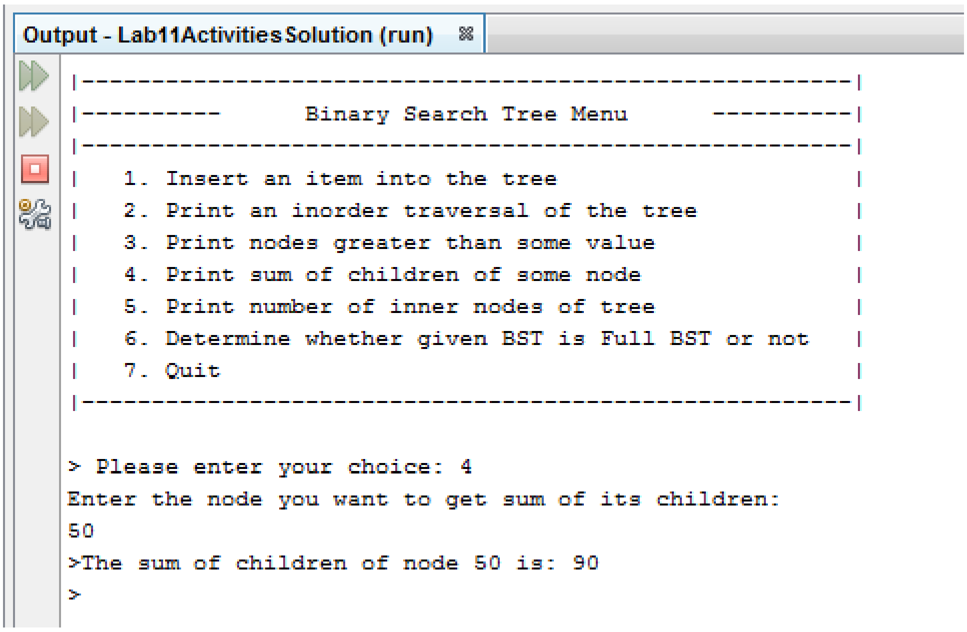
Sample Run2:



```
|-------------------------------------------------------|
|----------      Binary Search Tree Menu      ----------|
|-------------------------------------------------------|
|    1. Insert an item into the tree                    |
|    2. Print an inorder traversal of the tree          |
|    3. Print nodes greater than some value             |
|    4. Print sum of children of some node              |
|    5. Print number of inner nodes of tree             |
|    6. Determine whether given BST is Full BST or not  |
|    7. Quit                                            |
|-------------------------------------------------------|

> Please enter your choice: 4
Enter the node you want to get sum of its children:
35
>The sum of children of node 35 is: 33
>
```

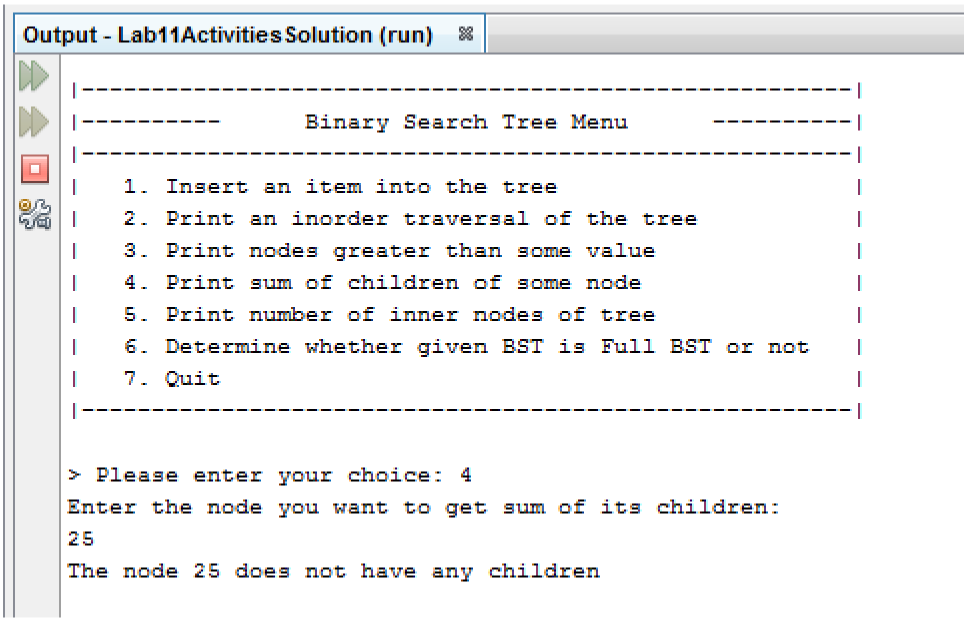
Sample Run3:



```
|-----------------------------------------------------------|
|----------      Binary Search Tree Menu       ----------|
|-----------------------------------------------------------|
|    1. Insert an item into the tree                        |
|    2. Print an inorder traversal of the tree              |
|    3. Print nodes greater than some value                 |
|    4. Print sum of children of some node                  |
|    5. Print number of inner nodes of tree                 |
|    6. Determine whether given BST is Full BST or not      |
|    7. Quit                                                |
|-----------------------------------------------------------|

> Please enter your choice: 4
Enter the node you want to get sum of its children:
50
>The sum of children of node 50 is: 90
>
```

Sample Run4:



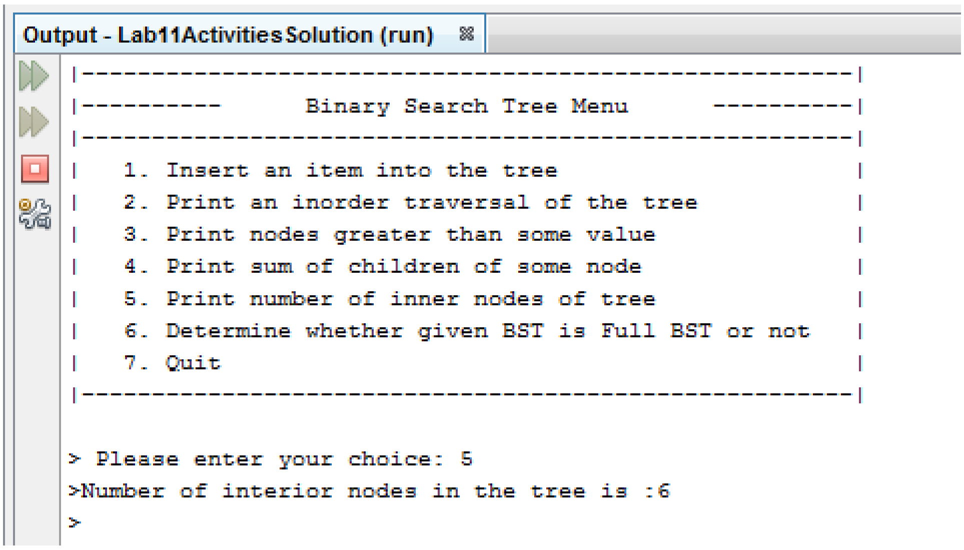
```
|-----------------------------------------------------------|
|----------      Binary Search Tree Menu       ----------|
|-----------------------------------------------------------|
|    1. Insert an item into the tree                        |
|    2. Print an inorder traversal of the tree              |
|    3. Print nodes greater than some value                 |
|    4. Print sum of children of some node                  |
|    5. Print number of inner nodes of tree                 |
|    6. Determine whether given BST is Full BST or not      |
|    7. Quit                                                |
|-----------------------------------------------------------|

> Please enter your choice: 4
Enter the node you want to get sum of its children:
25
The node 25 does not have any children
```

3. Write a method NonLeaves() in intBST class which counts and returns the number of interior nodes of BST.

Sample Run:

If we insert the nodes 50, 30, 25, 35, 33, 34, 60, 55 and 58 and choose the option 5, it will display the following output:



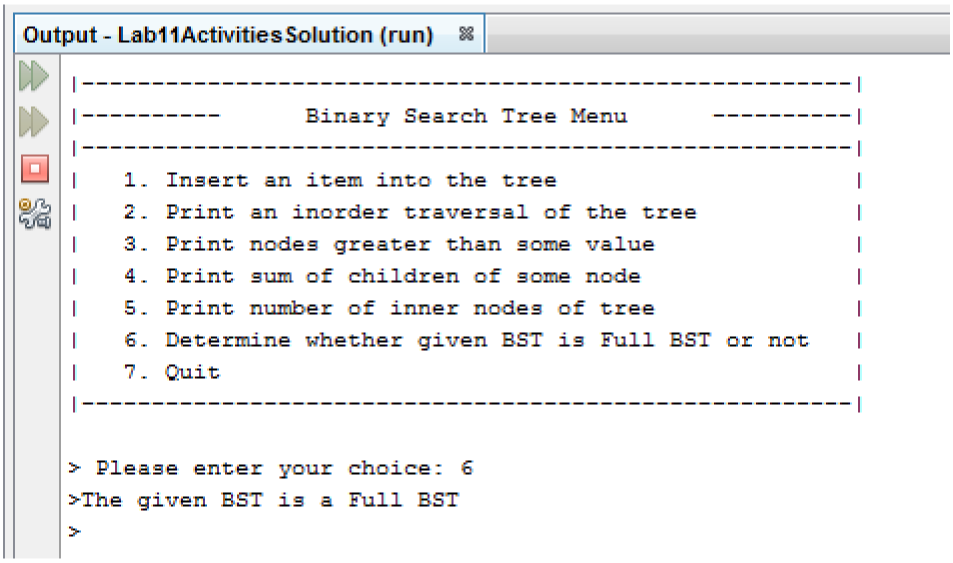
```
|-------------------------------------------------------|
|----------      Binary Search Tree Menu      ----------|
|-------------------------------------------------------|
|   1. Insert an item into the tree                     |
|   2. Print an inorder traversal of the tree           |
|   3. Print nodes greater than some value              |
|   4. Print sum of children of some node               |
|   5. Print number of inner nodes of tree              |
|   6. Determine whether given BST is Full BST or not   |
|   7. Quit                                             |
|-------------------------------------------------------|

> Please enter your choice: 5
>Number of interior nodes in the tree is :6
>
```

4. Write a method IsFullBST() in intBST class which returns 'true' if BST is a Full Binary Tree otherwise it should return 'false'.

Sample Run 1:

If we insert the nodes 50, 30, 25, 35 and 60and choose the option 6, it will display the following output:

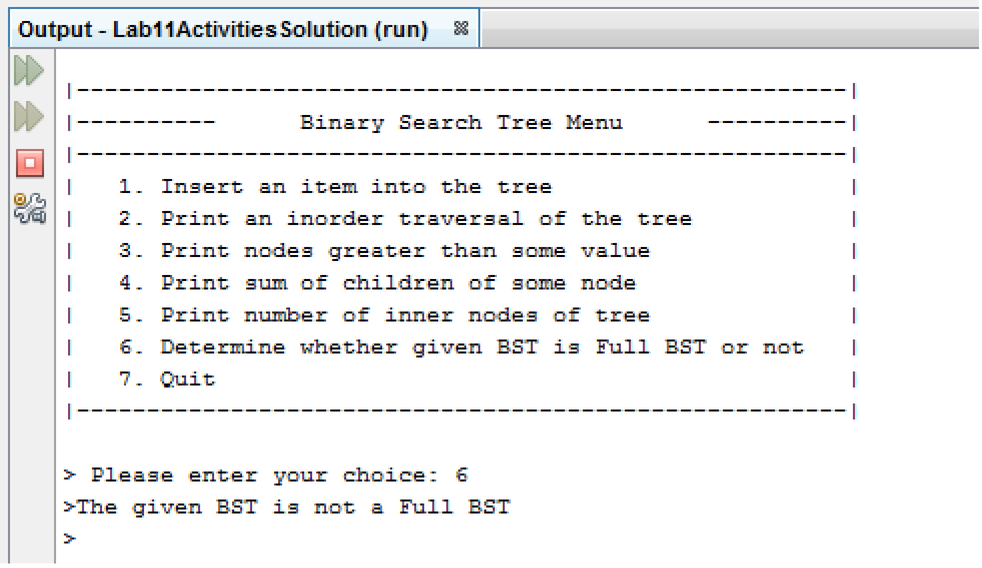


```
|------------------------------------------------------------|
|----------       Binary Search Tree Menu        ----------|
|------------------------------------------------------------|
|    1. Insert an item into the tree                         |
|    2. Print an inorder traversal of the tree               |
|    3. Print nodes greater than some value                  |
|    4. Print sum of children of some node                   |
|    5. Print number of inner nodes of tree                  |
|    6. Determine whether given BST is Full BST or not       |
|    7. Quit                                                 |
|------------------------------------------------------------|

> Please enter your choice: 6
>The given BST is a Full BST
>
```

Sample Run 2:

If we insert the nodes 50, 30, 25, 35, 33 and 60 and choose the option 6, it will display the following output:



```
|------------------------------------------------------------|
|----------       Binary Search Tree Menu        ----------|
|------------------------------------------------------------|
|    1. Insert an item into the tree                         |
|    2. Print an inorder traversal of the tree               |
|    3. Print nodes greater than some value                  |
|    4. Print sum of children of some node                   |
|    5. Print number of inner nodes of tree                  |
|    6. Determine whether given BST is Full BST or not       |
|    7. Quit                                                 |
|------------------------------------------------------------|

> Please enter your choice: 6
>The given BST is not a Full BST
>
```