```
::::::::::::::
Lab11Status.txt
::::::::::::::
Problem 1: compiles, runs correctly on all provided input
Problem 2: compiles, runs correctly on all provided input
Problem 3: compiles, runs correctly on all provided input
::::::::::::::
Lab11Conclusions.txt
::::::::::::::
I learned about Binary Search Trees (BSTs) and how they are mostly
used for efficiency in searching, insertion, and deletion operations
on a collection of elements. BSTs are a type of binary tree in which
each node has at most two children, and each node’s left child has a key
value less than its own key value, while its right child has a key value
greater than its own key value. This ordering property allows for fast
search and insertion operations, making BSTs useful for implementing
data structures like sets and maps. ::::::::::::::
MyBinarySearchTree.java
::::::::::::::
/*
 * Purpose: Data Structure and Algorithms Lab 11
 * Status: Complete and thoroughly tested
 * Last update: 4/17/23
 * Submitted:  4/17/23
 * Comment: Test suite and run sample attached.
 * Comment: I declare that this is entirely my own work
 * @author: Antonio Rosado
 * @version: 2023.04.17
 */
public class MyBinarySearchTree<T extends KeyedItem<KT>,
    KT extends Comparable<? super KT>>
    extends BinaryTreeBasis<T> {
    // inherits isEmpty(), makeEmpty(), getRootItem(), and
    // the use of the constructors from BinaryTreeBasis

    public MyBinarySearchTree() {
    }  // end default constructor

    public MyBinarySearchTree(T rootItem) {
        super(rootItem);
    }  // end constructor

    public void setRootItem(T newItem)
    throws UnsupportedOperationException {
        throw new UnsupportedOperationException();
    }  // end setRootItem

    public void insert(T newItem) {
        root = insertItem(root, newItem);
    }  // end insert

    /*
    * Retrieve Method (Iterative)
    * Retrieves an item from the binary search tree with the specified search key.
    * @param searchKey the search key of the item to be retrieved
    * @return the retrieved item, or null if the item is not found
    */
    public T retrieve(KT searchKey)
    {
        //implement iteratively
        TreeNode<T> currentNode = root;
        while (currentNode != null)
        {
            T currentItem = currentNode.getItem();
            if (searchKey.compareTo(currentItem.getKey()) == 0)
            {
                return currentItem;
            }

            else if (searchKey.compareTo(currentItem.getKey()) < 0)
            {
                currentNode = currentNode.getLeftChild();
            }

            else
            {
                currentNode = currentNode.getRightChild();
            }
        }
        return null;
    }  // end retrieve

    public void delete(KT searchKey) throws TreeException {
        root = deleteItem(root, searchKey);
    }  // end delete

    public void delete(T item) throws TreeException {
        root = deleteItem(root, item.getKey());
    }  // end delete

    protected TreeNode<T> insertItem(TreeNode<T> tNode, T newItem) {
        TreeNode<T> result;
        TreeNode<T> newSubtree;
        if (tNode == null) {
            // position of insertion found; insert after leaf
            // create a new node
            tNode = new TreeNode<T>(newItem, null, null);
            result = tNode;
        }  // end if
        else
        {   T nodeItem = tNode.getItem();

            // search for the insertion position

            if (newItem.getKey().compareTo(nodeItem.getKey()) < 0) {
                // search the left subtree
                newSubtree = insertItem(tNode.getLeftChild(), newItem);
                tNode.setLeftChild(newSubtree);
                result = tNode;
            }
            else { // search the right subtree
                newSubtree = insertItem(tNode.getRightChild(), newItem);
                tNode.setRightChild(newSubtree);
                result = tNode;
            }  // end if
        }
        return result;
    }  // end insertItem


    protected TreeNode<T> deleteItem(TreeNode<T> tNode, KT searchKey) {
        // Calls: deleteNode.

        TreeNode<T> newSubtree;
        if (tNode == null) {
            throw new TreeException("TreeException: Item not found");
        }
        else {
            T nodeItem = tNode.getItem();
            if (searchKey.compareTo(nodeItem.getKey()) == 0) {
                // item is in the root of some subtree
                tNode = deleteNode(tNode);  // delete the item
            }
            // else search for the item
            else if (searchKey.compareTo(nodeItem.getKey()) < 0) {
                // search the left subtree
                newSubtree = deleteItem(tNode.getLeftChild(), searchKey);
                tNode.setLeftChild(newSubtree);
            }
            else { // search the right subtree
                newSubtree = deleteItem(tNode.getRightChild(), searchKey);
                tNode.setRightChild(newSubtree);
            }  // end if
        }  // end if
        return tNode;
    }  // end deleteItem

    protected TreeNode<T> deleteNode(TreeNode<T> tNode) {
        // Algorithm note: There are four cases to consider:
        //   1. The tNode is a leaf.
        //   2. The tNode has no left child.
        //   3. The tNode has no right child.
        //   4. The tNode has two children.
        // Calls: findLeftmost and deleteLeftmost
        T replacementItem;
        TreeNode<T> result;
        // test for a leaf
        if ( (tNode.getLeftChild() == null) &&
                (tNode.getRightChild() == null) ) {
            result = null;
        }  // end if leaf
        else if (tNode.getLeftChild() == null) { // test for no left child
            result = tNode.getRightChild();
        }  // end if no left child
        else if (tNode.getRightChild() == null) { // test for no right child
            result = tNode.getLeftChild();
        }  // end if no right child

        // there are two children:
        // retrieve and delete the inorder successor
        else {
            replacementItem = findLeftmost(tNode.getRightChild());
            tNode.setItem(replacementItem);
            tNode.setRightChild(deleteLeftmost(tNode.getRightChild()));
            result = tNode;
        }  // end if
        return result;
    }  // end deleteNode

    /*
    * findLeftMost Method (Iterative)
    * Finds the leftmost TreeNode of the given TreeNode in a binary search tree.
    * @param tNode the TreeNode to search from
    * @return the leftmost TreeNode
    */
    protected T findLeftmost(TreeNode<T> tNode)
    {
        //implement iteratively
        TreeNode<T> currentNode = tNode;
        while (currentNode.getLeftChild() != null)
        {
            currentNode = currentNode.getLeftChild();
        }
        return currentNode.getItem();
    }  // end findLeftmost

    /*
    * deleteLeftMost Method (Iterative)
    * Deletes the leftmost TreeNode of the given TreeNode in a binary search tree.
    * @param tNode the TreeNode to delete from
    * @return the updated TreeNode after deletion
    */
    protected TreeNode<T> deleteLeftmost(TreeNode<T> tNode)
    {
        //implement iteratively
        TreeNode<T> currentNode = tNode;
        TreeNode<T> parent = null;

        while (currentNode.getLeftChild() != null)
        {
            parent = currentNode;
            currentNode = currentNode.getLeftChild();
        }

        if (parent == null)
        {
            return currentNode.getRightChild();
        }
        else
        {
            parent.setLeftChild(currentNode.getRightChild());
            return tNode;
        }
    }  // end deleteLeftmost
}  // end MyBinarySearchTree
::::::::::::::
MyBinarySearchTreePlus.java
::::::::::::::
/*
 * Purpose: Data Structure and Algorithms Lab 11
 * Status: Complete and thoroughly tested
 * Last update: 4/17/23
 * Submitted:  4/17/23
 * Comment: Test suite and run sample attached.
 * Comment: I declare that this is entirely my own work
 * @author: Antonio Rosado
 * @version: 2023.04.17
 */
public class MyBinarySearchTreePlus
    <T extends KeyedItem<KT>,
    KT extends Comparable<? super KT>>
    extends MyBinarySearchTree<T,KT>
    implements BSTPInterface<T,KT>
{

    /**
     * Returns a String representation of the elements in the tree traversed in-or
der.
```

```
     * @return a String representation of the tree elements in-order
     */
    @Override
    public String toStringInorder()
    {
        return inOrderHelper(root);
    }

    /**
     * Returns a String representation of the elements in the tree traversed pre-o
rder.
     * @return a String representation of the tree elements pre-order
     */
    @Override
    public String toStringPreorder()
    {
        return preOrderHelper(root);
    }

    /**
     * Returns a String representation of the elements in the tree traversed post-
order.
     * @return a String representation of the tree elements post-order
     */
    @Override
    public String toStringPostorder()
    {
        return postOrderHelper(root);
    }

    /**
     * Checks if the tree has the "odd-even" characteristic, which is defined as f
ollows:
     * At every level of the tree, the nodes must either have both children or no
children.
     * Additionally, if the level is even, the number of nodes in that level must
be even,
     * and if the level is odd, the number of nodes in that level must be odd.
     * @return true if the tree has the "odd-even" characteristic, false otherwise
     */
    @Override
    public boolean hasCharacteristic()
    {
        return hasCharacteristicHelper(root, 0);
    }

    /**
     * Private helper method to traverse the tree in-order and return a String rep
resentation of its elements.
     * @param node the root of the tree or subtree to traverse
     * @return a String representation of the tree or subtree in-order
     */
    private String inOrderHelper(TreeNode<T> node)
    {
        if(node == null)
        {
            return "";
        }
        String left = inOrderHelper(node.getLeftChild());
        String right = inOrderHelper(node.getRightChild());
        return left + node.getItem().toString() + " " + right;
    }
```

```
    /**
     * Private helper method to traverse the tree pre-order and return a String re
presentation of its elements.
     * @param node the root of the tree or subtree to traverse
     * @return a String representation of the tree or subtree pre-order
     */
    private String preOrderHelper(TreeNode<T> node)
    {
        if (node == null)
        {
            return "";
        }
        String left = preOrderHelper(node.getLeftChild());
        String right = preOrderHelper(node.getRightChild());
        return node.getItem().toString() + " " + left + right;
    }

    /**
    * Private helper method to traverse the tree post-order and return a String re
presentation of its elements.
    * @param node the root of the tree or subtree to traverse
    * @return a String representation of the tree or subtree post-order
    */
    private String postOrderHelper(TreeNode<T> node)
    {
        if (node == null)
        {
            return "";
        }
        String left = postOrderHelper(node.getLeftChild());
        String right = postOrderHelper(node.getRightChild());
        return left + right + node.getItem().toString() + " ";
    }

    /**
     * Checks if the tree has the "odd-even" characteristic, which is defined as f
ollows:
     * At every level of the tree, the nodes must either have both children or no
children.
     * Additionally, if the level is even, the number of nodes in that level must
be even,
     * and if the level is odd, the number of nodes in that level must be odd.
     * @return true if the tree has the "odd-even" characteristic, false otherwise
     */
    private boolean hasCharacteristicHelper(TreeNode<T> node, int level)
    {
        if (node == null)
        {
            return true;
        }
        if (level % 2 == 0)
        {
            if (node.getLeftChild() == null && node.getRightChild() == null
                    || node.getLeftChild() != null && node.getRightChild() == null
                    || node.getLeftChild() == null && node.getRightChild() != null
)
            {
                return hasCharacteristicHelper(node.getLeftChild(), level + 1) &&
hasCharacteristicHelper(node.getRightChild(), level + 1);
            }
            return false;
        }
        else
        {
            if (node.getLeftChild() != null && node.getRightChild() != null || nod
e.getLeftChild() == null && node.getRightChild() == null)
            {
                return hasCharacteristicHelper(node.getLeftChild(), level + 1) &&
hasCharacteristicHelper(node.getRightChild(), level + 1);
            }
            return false;
        }
    }

}
::::::::::::::
Item.java
::::::::::::::
/*
 * Purpose: Data Structure and Algorithms Lab 11
 * Status: Complete and thoroughly tested
 * Last update: 4/17/23
 * Submitted:  4/17/23
 * Comment: Test suite and run sample attached.
 * Comment: I declare that this is entirely my own work
 * @author: Antonio Rosado
 * @version: 2023.04.17
 */
public class Item <KT extends Comparable<? super KT>> extends KeyedItem<KT>
{

    private boolean assocboolean;
    private String assocstring;
    public Item(KT key, boolean assocboolean, String assocstring)
    {
        super(key);
        this.assocboolean = assocboolean;
        this.assocstring = assocstring;
    }

    public boolean isAssocboolean()
    {
        return assocboolean;
    }

    public void setAssocboolean(boolean assocboolean)
    {
        this.assocboolean = assocboolean;
    }

    public String getAssocstring()
    {
        return assocstring;
    }

    public void setAssocstring(String assocstring)
    {
        this.assocstring = assocstring;
    }

    @Override
    public String toString()
    {
        return "{" + getKey() + "/" + assocboolean + "/" + assocstring + "}";
    }

}::::::::::::::
Lab11P3Driver.java
::::::::::::::
/*
 * Purpose: Data Structure and Algorithms Lab 11
 * Status: Complete and thoroughly tested
 * Last update: 4/17/23
 * Submitted:  4/17/23
 * Comment: Test suite and run sample attached.
 * Comment: I declare that this is entirely my own work
 * @author: Antonio Rosado
 * @version: 2023.04.17
 */
import java.io.IOException;
import java.io.BufferedReader;
import java.io.InputStreamReader;
public class Lab11P3Driver
{
    private static MyBinarySearchTreePlus tree = new MyBinarySearchTreePlus();
    private static BufferedReader stdin = new BufferedReader(new InputStreamReader
(System.in));
    public static void main (String[] args) throws IOException
    {
        boolean exit = false;
        while (!exit)
        {
            System.out.println("Select from the following menu: \n"
                               + "0. Exit the program \n"
                               + "1. Search for item in BST \n"
                               + "2. Insert item in BST \n"
                               + "3. Delete item from BST \n"
                               + "4. Display content of BST in-order \n"
                               + "5. Display content of BST in post-order \n"
                               + "6. Display content of BST in pre-order \n"
                               + "7. Check if BST has required characteristic \n")
;

            System.out.print("Make your menu selection now: " );
            int input = Integer.parseInt(stdin.readLine());
            System.out.println(input);
            // possible cases for initial input
            switch (input)
            {
            case 0:
                System.out.println("Exiting program... good bye");
                exit = true;
                break;

            case 1:
                searchItem();
                break;

            case 2:
                insertItem();
                break;

            case 3:
                deleteItem();
                break;
            case 4:
                inOrderDisplay();
                break;

            case 5:
                postOrderDisplay();
                break;

            case 6:
                preOrderDisplay();
                break;

            case 7:
                characteristic();
                break;

            default:
                System.out.println("Invalid option, please try again.\n");
                break;
            }
        }
    }
    private static void searchItem() throws IOException
    {
        System.out.print("Enter the key of the item to search for: ");
        String key = stdin.readLine();
        System.out.println(key);
        KeyedItem item = tree.retrieve(key);
        if (item == null)
        {
            System.out.println("Item with key ’" + key + "’ not found in BST.");
        }
        else
        {
            System.out.println("Item with key ’" + key + "’ found in BST:\n" + ite
m.toString());
        }
    }

    private static void insertItem() throws IOException
    {
        System.out.print("Enter the key of the item to insert: ");
        String key = stdin.readLine();
        System.out.println(key);

        System.out.print("Enter associated boolean: ");
        String bool = stdin.readLine();
        System.out.println(bool);
        boolean assocboolean;
        if(bool.equalsIgnoreCase("true"))
        {
            assocboolean = true;
        }
        else
        {
            assocboolean = false;
        }
        System.out.print("Enter associated string: ");
        String assocstring = stdin.readLine();
        System.out.println(assocstring);
        KeyedItem existingItem = tree.retrieve(key);
```

```
        if (existingItem != null)
        {
            System.out.println("Item with key ’" + key + "’ already exists in BST
and cannot be inserted again.");
        }
        else
        {
            tree.insert(new Item(key, assocboolean, assocstring));
            System.out.println("Item " + assocstring + " inserted into BST with ke
y " + key);
        }
    }

    private static void deleteItem() throws IOException
    {
        System.out.print("Enter the key of the item to delete: ");
        String key = stdin.readLine();
        System.out.println(key);
        KeyedItem existingItem = tree.retrieve(key);
        if(existingItem != null)
        {
            System.out.println("Item " + "’" + key + "’" + " deleted.");
            tree.delete(existingItem);
        }
        else
        {
            System.out.println("Item does not exist, cannot delete a non-existent
item!");
        }
    }

    private static void characteristic()
    {
        System.out.println("Checking if BST has required characteristic...");
        if(!(tree.isEmpty()))
        {
            boolean hasCharacteristic = tree.hasCharacteristic();
            if (hasCharacteristic)
            {
                System.out.println("The BST has the required characteristic.");
            }
            else
            {
                System.out.println("The BST does NOT have the required characteris
tic!");
            }
        }

        else
        {
            System.out.println("The BST is empty.");
        }
    }

    private static void inOrderDisplay()
    {
        if(tree.isEmpty())
        {
            System.out.println("Empty BST - nothing to display.");
        }

        else
        {
            System.out.println("Content of the BST in-order:\n" + tree.toStringIno
rder());

        }
    }

    private static void postOrderDisplay()
    {
        if(tree.isEmpty())
        {
            System.out.println("Empty BST - nothing to display.");
        }

        else
        {
            System.out.println("Content of the BST post-order:\n" + tree.toStringP
ostorder());
        }
    }

    private static void preOrderDisplay()
    {
        if(tree.isEmpty())
        {
            System.out.println("Empty BST - nothing to display.");
        }

        else
        {
            System.out.println("Content of the BST pre-order:\n" + tree.toStringPr
eorder());
        }
    }
}

::::::::::::::
Lab11P3Output.txt
::::::::::::::
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Empty BST - nothing to display.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Empty BST - nothing to display.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Empty BST - nothing to display.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST is empty.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 1
Enter the key of the item to search for: 6
Item with key '6' not found in BST.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 13
Enter associated boolean: true
Enter associated string: shoe
Item shoe inserted into BST with key 13
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
```

```
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST has the required characteristic.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 20
Item does not exist, cannot delete a non-existent item!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 13
Item ’13’ deleted.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST is empty.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 33
Enter associated boolean: false
Enter associated string: sock
Item sock inserted into BST with key 33
Select from the following menu:
0. Exit the program
1. Search for item in BST
```

```
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST has the required characteristic.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 8
Enter associated boolean: false
Enter associated string: short
Item short inserted into BST with key 8
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST has the required characteristic.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 27
Enter associated boolean: true
Enter associated string: tie
Item tie inserted into BST with key 27
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic
```

```
Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 20
Enter associated boolean: false
Enter associated string: hat
Item hat inserted into BST with key 20
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 5
Enter associated boolean: true
Enter associated string: shirt
Item shirt inserted into BST with key 5
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{20/false/hat} {27/true/tie} {33/false/sock} {5/true/shirt} {8/false/short}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{20/false/hat} {27/true/tie} {5/true/shirt} {8/false/short} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {27/true/tie} {20/false/hat} {8/false/short} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 40
Item does not exist, cannot delete a non-existent item!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 25
Enter associated boolean: false
Enter associated string: shoe
Item shoe inserted into BST with key 25
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 5
Enter associated boolean: false
Enter associated string: vest
Item with key ’5’ already exists in BST and cannot be inserted again.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 2
Enter associated boolean: true
Enter associated string: cap
Item cap inserted into BST with key 2
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 11
Enter associated boolean: false
Enter associated string: hat
Item hat inserted into BST with key 11
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {2/true/cap} {20/false/hat} {25/false/shoe} {27/true/tie} {33/false
/sock} {5/true/shirt} {8/false/short}
```

```
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{11/false/hat} {2/true/cap} {25/false/shoe} {20/false/hat} {27/true/tie} {5/true/s
hirt} {8/false/short} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {27/true/tie} {20/false/hat} {2/true/cap} {11/false/hat} {25/false
/shoe} {8/false/short} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 4
Enter associated boolean: false
Enter associated string: vest
Item vest inserted into BST with key 4
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
```

```
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 24
Item does not exist, cannot delete a non-existent item!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {2/true/cap} {20/false/hat} {25/false/shoe} {27/true/tie} {33/false
/sock} {4/false/vest} {5/true/shirt} {8/false/short}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{11/false/hat} {2/true/cap} {25/false/shoe} {20/false/hat} {27/true/tie} {4/false/
vest} {5/true/shirt} {8/false/short} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {27/true/tie} {20/false/hat} {2/true/cap} {11/false/hat} {25/false
/shoe} {8/false/short} {5/true/shirt} {4/false/vest}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 4
Item '4' deleted.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 16
Enter associated boolean: true
Enter associated string: glove
Item glove inserted into BST with key 16
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 1
Enter associated boolean: false
Enter associated string: pant
Item pant inserted into BST with key 1
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
```

```
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 1
Enter the key of the item to search for: 9
Item with key ’9’ not found in BST.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 1
Enter the key of the item to search for: 20
Item with key ’20’ found in BST:
{20/false/hat}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 1
Enter the key of the item to search for: 11
Item with key ’11’ found in BST:
{11/false/hat}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 19
Enter associated boolean: false
Enter associated string: belt
Item belt inserted into BST with key 19
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{1/false/pant} {11/false/hat} {16/true/glove} {19/false/belt} {2/true/cap} {20/false/hat} {25/false/shoe} {27/true/tie} {33/false/sock} {5/true/shirt} {8/false/short}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{1/false/pant} {19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {25/false/shoe} {20/false/hat} {27/true/tie} {5/true/shirt} {8/false/short} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {27/true/tie} {20/false/hat} {2/true/cap} {11/false/hat} {1/false/pant} {16/true/glove} {19/false/belt} {25/false/shoe} {8/false/short} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
```

```
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 15
Enter associated boolean: true
Enter associated string: tie
Item tie inserted into BST with key 15
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 1
Item ’1’ deleted.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {15/true/tie} {16/true/glove} {19/false/belt} {2/true/cap} {20/fals
e/hat} {25/false/shoe} {27/true/tie} {33/false/sock} {5/true/shirt} {8/false/short
}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{15/true/tie} {19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {25/fals
e/shoe} {20/false/hat} {27/true/tie} {5/true/shirt} {8/false/short} {33/false/sock
}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {27/true/tie} {20/false/hat} {2/true/cap} {11/false/hat} {16/true/
glove} {15/true/tie} {19/false/belt} {25/false/shoe} {8/false/short} {5/true/shirt
}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 8
Item ’8’ deleted.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {15/true/tie} {16/true/glove} {19/false/belt} {2/true/cap} {20/fals
e/hat} {25/false/shoe} {27/true/tie} {33/false/sock} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{15/true/tie} {19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {25/fals
e/shoe} {20/false/hat} {27/true/tie} {5/true/shirt} {33/false/sock}
```

```
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {27/true/tie} {20/false/hat} {2/true/cap} {11/false/hat} {16/true/
glove} {15/true/tie} {19/false/belt} {25/false/shoe} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 27
Item ’27’ deleted.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {15/true/tie} {16/true/glove} {19/false/belt} {2/true/cap} {20/fals
e/hat} {25/false/shoe} {33/false/sock} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{15/true/tie} {19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {25/fals
e/shoe} {20/false/hat} {5/true/shirt} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {20/false/hat} {2/true/cap} {11/false/hat} {16/true/glove} {15/tru
e/tie} {19/false/belt} {25/false/shoe} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 15
Item ’15’ deleted.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 19
Enter associated boolean: false
Enter associated string: tie
Item with key ’19’ already exists in BST and cannot be inserted again.
Select from the following menu:
```

```
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 42
Enter associated boolean: true
Enter associated string: shoe
Item shoe inserted into BST with key 42
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 28
Enter associated boolean: false
Enter associated string: shirt
Item shirt inserted into BST with key 28
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {16/true/glove} {19/false/belt} {2/true/cap} {20/false/hat} {25/fal
se/shoe} {28/false/shirt} {33/false/sock} {42/true/shoe} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {28/false/shirt} {25/f
alse/shoe} {20/false/hat} {42/true/shoe} {5/true/shirt} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
```

```
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {20/false/hat} {2/true/cap} {11/false/hat} {16/true/glove} {19/fal
se/belt} {25/false/shoe} {28/false/shirt} {5/true/shirt} {42/true/shoe}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 14
Item does not exist, cannot delete a non-existent item!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {16/true/glove} {19/false/belt} {2/true/cap} {20/false/hat} {25/fal
se/shoe} {28/false/shirt} {33/false/sock} {42/true/shoe} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {28/false/shirt} {25/f
alse/shoe} {20/false/hat} {42/true/shoe} {5/true/shirt} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {20/false/hat} {2/true/cap} {11/false/hat} {16/true/glove} {19/fal
se/belt} {25/false/shoe} {28/false/shirt} {5/true/shirt} {42/true/shoe}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 2
Enter the key of the item to insert: 33
Enter associated boolean: true
Enter associated string: belt
Item with key ’33’ already exists in BST and cannot be inserted again.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 21
Item does not exist, cannot delete a non-existent item!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {16/true/glove} {19/false/belt} {2/true/cap} {20/false/hat} {25/fal
se/shoe} {28/false/shirt} {33/false/sock} {42/true/shoe} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {28/false/shirt} {25/f
alse/shoe} {20/false/hat} {42/true/shoe} {5/true/shirt} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {20/false/hat} {2/true/cap} {11/false/hat} {16/true/glove} {19/fal
se/belt} {25/false/shoe} {28/false/shirt} {5/true/shirt} {42/true/shoe}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 13
Item does not exist, cannot delete a non-existent item!
```

```
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{11/false/hat} {16/true/glove} {19/false/belt} {2/true/cap} {20/false/hat} {25/fal
se/shoe} {28/false/shirt} {33/false/sock} {42/true/shoe} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{19/false/belt} {16/true/glove} {11/false/hat} {2/true/cap} {28/false/shirt} {25/f
alse/shoe} {20/false/hat} {42/true/shoe} {5/true/shirt} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {20/false/hat} {2/true/cap} {11/false/hat} {16/true/glove} {19/fal
se/belt} {25/false/shoe} {28/false/shirt} {5/true/shirt} {42/true/shoe}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 3
Enter the key of the item to delete: 11
Item ’11’ deleted.
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 4
Content of the BST in-order:
{16/true/glove} {19/false/belt} {2/true/cap} {20/false/hat} {25/false/shoe} {28/fa
lse/shirt} {33/false/sock} {42/true/shoe} {5/true/shirt}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 5
Content of the BST post-order:
{19/false/belt} {16/true/glove} {2/true/cap} {28/false/shirt} {25/false/shoe} {20/
false/hat} {42/true/shoe} {5/true/shirt} {33/false/sock}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 6
Content of the BST pre-order:
{33/false/sock} {20/false/hat} {2/true/cap} {16/true/glove} {19/false/belt} {25/fa
lse/shoe} {28/false/shirt} {5/true/shirt} {42/true/shoe}
Select from the following menu:
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 7
Checking if BST has required characteristic...
The BST does NOT have the required characteristic!
Select from the following menu:
```

```
0. Exit the program
1. Search for item in BST
2. Insert item in BST
3. Delete item from BST
4. Display content of BST in-order
5. Display content of BST in post-order
6. Display content of BST in pre-order
7. Check if BST has required characteristic

Make your menu selection now: 0
Exiting program... good bye
```