```
+ main()
                                      DSABinarySearchTree
- root: DSATreeNode
+ insert(String key, Object value)
+ find(String key): Object findRec(DSATreeNode root)
+ delete(String key)
+ height(): int heightRec(DSATreeNode root)
+ min(): int minRec(DSATreeNode root)
+ max(): int maxRec(DSATreeNode root)
+ count(): int countRec(DSATreeNode root)
+ balance(): int balancePriv(DSATreeNode root)
+ traversePreorder: DSAQueue traversePreorderRec(DSATreeNode root, DSAQueue bstQueue)
+ traverseInorder: DSAQueue traverseInorderRec(DSATreeNode root, DSAQueue bstQueue)
+ traversePostorder: DSAQueue traversePostorderRec(DSATreeNode root, DSAQueue bstQueue)
+ display()

    insertRec(String key, Object value, DSATreeNode currNode): DSATreeNode updateNode

findRec(String key, DSATreeNode currNode): Object value

    deleteRec(String key, DSATreeNode currNode): DSATreeNode updateNode

    deleteNode(String key, DSATreeNode delNode): DSATreeNode updateNode

promoteSuccessor(DSATreeNode currNode): succNode
heightRec(DSATreeNode currNode): int currHgt
- minRec(DSATreeNode currNode): int minKev
- maxRec(DSATreeNode currNode): int maxKey
countRec(DSATreeNode currNode): int numElems
balancePriv(DSATreeNode currNode): double pctge

    traversePreorderRec(DSATreeNode currNode, DSAQueue bstQueue)

    traverseInorderRec(DSATreeNode currNode, DSAQueue bstQueue)

    traversePostorderRec(DSATreeNode currNode, DSAQueue bstQueue)

    displayRec(DSATreeNode currNode, int currLvl)

- currLvlToSpaces(int currLvl): String spaces
                                          DSATreeNode
                       - key: String
                       - value: Object
                       - leftChild: DSATreeNode
                       - rightChild: DSATreeNode
                       + setKey(String inKey)
                       + setValue(Object inValue)
                       + setLeft(DSATreeNode inLeftChild)
                       + setRight(DSATreeNode inRightChild)
                       + getKey(): String key
                       + getValue(): Object value
                       + getLeft(): DSATreeNode leftChild
```

+ getRight(): DSATreeNode rightChild

DSABinarySearchTreeTestHarness