```
startOfFile
  TreeNode class implementation:
      int val
      TreeNode left
      TreeNode right

    In the TreeNode constructor (int x)
      set val equals x

    End constructor

    In main method
      arrayToBST arrayToBST = new arrayToBST
      TreeNode root
      int array = {//sample array};

      set root equals to arrayToBST method (array)
      print out "Preorder traversal of constructed BST "
      call Preorder method (root)

    End main method
  END TreeNode implementation

  arrayToBST class implementation:

    In method -type-TreeNode arrayToBST(int array)
        if array length equals 0
                return Null
        Else
                return treeFromArray(array, 0, array.length - 1)

    END arrayToBST method

    In method -type-TreeNode treeFromArray(int array, firstIndex, lastIndex)
        if firstIndex > lastIndex
                return Null
        Else
                int midpoint = firstIndex + (lastIndex-firstIndex) / 2
                -type-TreeNode node = new TreeNode(array[midpoint])
                node.left equals recursive call treeFromArray(arr,firstIndex,midpoint-1)
                node.right equals recursive call treeFromArray(arr,midpoint+1, lastIndex)
                return node
    END treeFromArray method
  In Preorder method (-type-TreeNode node )
     if node equals Null
          return

  print out node.val + " "
  recursive call Preorder(node.left)
  recursive call Preorder(node.right)
  END Preorder method
 END arrayToBST implementation
endOfFile
```