Homework Turnin

Email: rgalanos@fcps.edu
Section: 6G
Course: TJHSST APCS 2016-17
Assignment: 04-01

Receipt ID: f6ddd45a505fd83aed8b33e6d16a4e17

Turnin Successful!

The following file(s) were received:

ListLab1.java   (2892 bytes)

```java
//name:     date:
import java.util.*;
public class ListLab1
{
    public static void main(String[] args)
    {
        ListNode head = new ListNode("hello", null);
        head = new ListNode("foo", head);
        head = new ListNode("boo", head);
        head = new ListNode("nonsense", head);
        head = new ListNode("computer",
                new ListNode("science",
                    new ListNode("java",
                        new ListNode("coffee", head)
                    )
                )
            );
        print(head);
        print(head);

        /**** uncomment the code below for ListLab1 Extension  ****/

        System.out.println("First = " + first(head));
        System.out.println("Second = " + second(head));
        ListNode p = pointerToLast(head);
        System.out.println("Pointer to Last = " + p.getValue()+ " at " + p);
        ListNode c = copyOfLast(head);
        System.out.println("Copy of Last =    " + c.getValue()+ " at " + c);

        Scanner in = new Scanner(System.in);
        System.out.print("Insert what? ");
        String x = in.next();
        head = insertFirst(head, x);
        head = insertLast(head, x);
        print(head);
    }
    public static void print(ListNode head)
    {
        System.out.print("[");
        while(head != null)
        {
            System.out.print(head.getValue());
            head = head.getNext();
            if(head != null)
                System.out.print(", ");
        }
        System.out.println("]");
    }
    private static ListNode copyNode(ListNode arg)
    {
        return new ListNode(arg.getValue(), arg.getNext());
```

```java
    }
    private static ListNode copyList(ListNode arg)
    {
        if(arg.getNext()==null)
            return copyNode(arg);
        else
            return new ListNode(arg.getValue(), copyList(arg.getNext()));
    }
    private static ListNode rest(ListNode h)
    {
        return copyList(h.getNext());
    }
    public static Object first(ListNode head)
    {
        if(head == null)
            return null;
        else
            return head.getValue();
    }
    public static Object second(ListNode head)
    {
        if(head == null)
            return null;
        else if(head.getNext() == null)
            return null;
        else
            return first(head.getNext());
    }
    public static ListNode pointerToLast(ListNode head)
    {
        if(head == null)
            return null;
        else
        {
            while(head.getNext()!=null)
                head = head.getNext();
            return head;
        }
    }
    public static ListNode copyOfLast(ListNode head)
    {
        head = pointerToLast(head);
        return new ListNode(head.getValue(),head.getNext());
    }
    public static ListNode insertFirst(ListNode head, Object arg)
    {
        head = new ListNode(arg, head);
        return head;
    }
    public static ListNode insertLast(ListNode head, Object arg)
    {
        pointerToLast(head).setNext(new ListNode(arg, null));
        return head;
    }
}
```