

IntListA

List implementation without a sentinel value

IntListA

```
import java.io.*;
import java.util.*;

public class IntListA {
//Implements a singly linked list of ints without a sentinel node

    private class Node {
        private int data;
        private Node next;
        private Node(int d, Node n) {
            data = d;
            next = n;
        }
    }

    private Node head;
```

IntListA

```
public IntListA() {  
    head = null; //no sentinel node  
}  
  
public boolean empty() {  
    return head == null;  
}  
public void insertFirst(int d) {  
    head = new Node(d, head);  
}  
  
public void insertLast(int d) {  
    COMPLETED IN CLASS  
}
```

IntListA

```
public int minI() {  
    //PRE: !empty()  
    //returns the smallest int in the list  
    //the implementation must be iterative  
  
    int min = head.data;  
    Node temp = head.next;  
    while (temp != null) {  
        if (min > temp.data)  
            min = temp.data;  
        temp = temp.next;  
    }  
    return min;  
}
```

IntListA

```
public int minR() {  
    //PRE: !empty()  
    //returns the smallest int in the list  
    //the implementation must be recursive  
        return minR(head);  
}
```

```
public int minR(Node h) {  
    //PRE: h != null  
    //returns the minimum int in the list beginning at h  
    //the implementation must be recursive  
        if (h.next == null) return h.data;  
        int min = minR(h.next);  
        return h.data < min ? h.data : min;  
}
```

IntListA

```
public String toString() {  
    if (head == null) return "[]";  
    String s = "[" + head.data;  
    Node temp = head.next;  
    while (temp != null) {  
        s = s + ',' + temp.data;  
        temp = temp.next;  
    }  
    return s + "];"  
}
```

IntListA: Exercises

- Implement a recursive function to find the sum of the ints in the list
- Implement a recursive function that returns a string representation of the list
- Implement a remove function