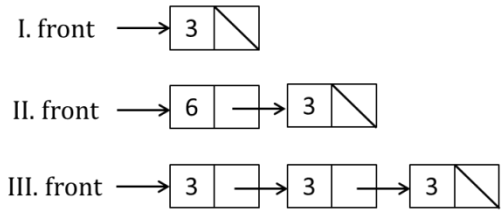
Linked List Test 1

Consider the following data fields and methods.

```
private ListNode front;

public void processList(Integer val)
{
   ListNode temp, prev;

   while (front != null && front.getValue().equals(val))
   {
      front = front.getNext();
   }

   if (front == null)
       return;

   prev = front;
   temp = front.getNext();

   while (temp != null)
   {
       if (temp.getValue().equals(val))
       {
           prev.setNext(temp.getNext());
       }
       else
       {
           prev = temp;
       }
       temp = temp.getNext();
    }
 }
```

1. Which of the following best describes what `processList` does?
   A) It removes all consecutive nodes at the front of the list with value `val`.
   B) It removes the first occurrence of the node with value `val`.
   C) It removes all nodes with value `val`.
   D) It removes all nodes with value `val` except the last such node.

2. Consider modifying `processList` by removing the `if` statement following the first `while` loop. For which of the following linked lists would the call `processList(new Integer(3))` result in a run-time error?



A) I only
B) III only
C) I and II only
D) I and III only

3. Consider the following method:

```
public ListNode mystery(ListNode head)
{
   ListNode r = null, p = null;

   while (head != null)
   {
      r = head.getNext();
      head.setNext(p);
      p = head;
      head = r;
   }
   return p;
}
```

If head refers to the first node of a linked list with five nodes, A → B → C → D → E, which of the following lists is returned by `mystery(head)`?

A) B → C → D → E
B) A → B → C → D
C) A → B → C → D → E
D) E → D → C → B → A

4. Consider the following method.

```
public ListNode mystery(ListNode node)
{
   if (node == null)
      return null;
   else
      return new ListNode(node.getValue(), mystery(node.getNext()));
}
```

A) Always returns null
B) Creates and returns a copy of the given list
C) Creates and returns a reversed copy of the given list
D) Creates and returns a copy of the first node of the given list

5. Consider the following class:

```
public class MyLinkedList
{
   private ListNode front;

   public MyLinkedList()
   {
      front = null;
   }

   public void addLast(Object val)
   {
      if (front == null)
      {
         front = new ListNode(val, null);
      }
      else
      {
         < missing code >
      }
   }
}
```

The `lastNode` method should add new values to the end of the linked List. Which of the following code segments can replace *< missing code >?*

A)
```
while (front != null)
   {
      front = front.getNext();
   }
   front = new ListNode(val, null);
```

B)
```
while (front.getNext() != null)
   {
      front = front.getNext();
   }
   front.setNext(new ListNode(val, null);
```

C)
```
ListNode p = front;
   while (p != null)
   {
      p = p.getNext();
   }
   p = new ListNode(val, null);
```

D)
```
ListNode p = front;
   while (p.getNext() != null)
   {
      p = p.getNext();
   }
   p.setNext(new ListNode(val, null));
```