

Quiz

Tuesday, October 9, 2018 3:23 PM

CSE143 Section #4 Problems

Fill in the "code" column in the following tables providing a solution that will turn the "before" picture into the "after" picture by modifying links between the nodes shown. You are not allowed to change any existing node's data field value and you are not allowed to construct any new nodes, but you are allowed to declare and use variables of type ListNode (often called "temp" variables). You are limited to at most two variables of type ListNode for each of the subproblems below.

You are writing code for the ListNode class discussed in lecture:

```
public class ListNode {  
    public int data; // data stored in this node  
    public ListNode next; // link to next node in the list  
  
    <constructors>  
}
```

As in the lecture examples, all lists are terminated by null and the variables p and q have the value null when they do not point to anything.

1. Basic list manipulation.

before	after	code
p->[1]->[2]->[3] q	p->[1]->[2] q->[3]	q = p.next; p.next=null;
p->[1]->[2] q->[3]	p->[2] q->[3]->[1]	q.next=p; p=p.next; q.next.next=null;
p->[1]->[2] q->[3]	p q->[1]->[2]->[3]	p.next=q; q=p; p=null;
p->[1]->[2] q->[3]	p->[1]->[3]->[2] q	q.next=p.next; p.next=q; q=null;
p->[1]->[2] q->[3]->[4]	p->[1]->[3]->[4]->[2] q	q.next.next=p.next; p.next=q; q=null;
p->[5]->[4]->[3] p->[1]->[2] q->[3]	p->[4]->[3]->[5] p->[3]->[1]->[2] q	p.next.next=p; p=p.next; p.next.next=null;
		q.next=p; p=q; q=null;

2. Intermediate list manipulation.

before	after	code
p->[1]->[2] q->[3]->[4]	p->[1]->[3]->[2]->[4] q	p.next.next=q.next; q.next=p.next; p.next=q; q=null;
p->[1] q->[2]->[3]	p->[3] q->[1]->[2]	p.next=q; q=p; p=q.next.next; q.next.next=null;
p->[1]->[2] q->[3]->[4]	p->[2]->[4] q->[3]->[1]	p.next.next=q.next; q.next=p; p=q.next.next; q.next.next=null;
p->[1]->[2]->[3]	p->[2] q	p: 124; q: 34 q.next=p; p=q.next.next; q.next.next=null; q=p

$q \rightarrow [3] \rightarrow [4]$	$q \rightarrow [3] \rightarrow [1]$	$q \rightarrow [3] \rightarrow [4]$	$q = 3$ $p = 2$ $q = 2$
$p \rightarrow [1] \rightarrow [2] \rightarrow [3]$	$p \rightarrow [2]$	$q = p;$ $p = q.next.next;$ $q.next.next = null;$ $p.next = null;$	
q	$q \rightarrow [1] \rightarrow [3]$	$q = p.next; q.next = q.next.next;$ $p.next = null;$	
$p \rightarrow [5] \rightarrow [4] \rightarrow [3]$	$p \rightarrow [4] \rightarrow [5] \rightarrow [3]$	$ListNode temp = new ListNode(3);$ $temp = p;$ $p = temp.next;$ $temp.next = temp.next.next;$ $temp.next.next = null;$	

3. List manipulation involving construction. Unlike in the other problems, you are to construct a list node for these problems. Remember that the `ListNode` class has a 2-argument constructor that takes the data and link as parameters.

before	after	code
$p \rightarrow [1] \rightarrow [2]$	$p \rightarrow [1] \rightarrow [2] \rightarrow [3]$	$p.next.next = new ListNode(3);$
$p \rightarrow [1] \rightarrow [2]$	$p \rightarrow [3] \rightarrow [1] \rightarrow [2]$	$p = new ListNode(3, p);$
$p \rightarrow [1] \rightarrow [2]$	$p \rightarrow [1] \rightarrow [3] \rightarrow [2]$	$p.next =$ $new ListNode(3, p.next);$

4. Advanced list manipulation.

before	after	code
$p \rightarrow [5] \rightarrow [4] \rightarrow [3]$	$p \rightarrow [3] \rightarrow [4] \rightarrow [5]$	$ListNode temp = new ListNode();$ $temp = p.next.next; \quad \rightarrow p.next = null;$ $p.next.next = null; \quad \rightarrow temp.next.next = p;$ $temp.next = p.next; \quad \rightarrow p = temp;$ $q.next.next = p.next; \quad \rightarrow p.next = q.next;$ $q.next.next.next = p; \quad \rightarrow q.next = null;$ $p = q; \quad \rightarrow p = q.next;$ $q = q.next; \quad \rightarrow q = null;$
$p \rightarrow [1] \rightarrow [2]$	$p \rightarrow [3] \rightarrow [2] \rightarrow [1]$	
$q \rightarrow [3] \rightarrow [4]$	$q \rightarrow [4]$	
$p \rightarrow [1] \rightarrow [2]$	$p \rightarrow [1] \rightarrow [3] \rightarrow [5]$	
$q \rightarrow [3] \rightarrow [4] \rightarrow [5]$	$q \rightarrow [2] \rightarrow [4]$	
$p \rightarrow [1] \rightarrow [2] \rightarrow [3]$	$p \rightarrow [2] \rightarrow [1] \rightarrow [4]$	
$q \rightarrow [4] \rightarrow [5]$	$q \rightarrow [3] \rightarrow [5]$	
$p \rightarrow [1] \rightarrow [2] \rightarrow [3] \quad 45$	$p \rightarrow [5] \rightarrow [4] \rightarrow [3] \rightarrow [2]$	$p.next.next.next = q;$ $q = p;$ $p = q.next.next.next.next;$ $q.next.next.next.next = null;$ $p.next = q.next.next.next;$ $q.next.next.next = null;$ $p.next.next = q.next.next;$ $q.next.next = null;$ $p.next.next.next = q.next;$ $q.next = null;$
$q \rightarrow [4] \rightarrow [5]$	$q \rightarrow [1]$	