More linked memory

1/18/16
Announcements

• HW due Wednesday: replaceAll tests and methods

• Reading: Chapter 4 for Friday
Recall: Linked implementation of Bag

- “head” points to first node

Null next pointer means “end of the list”
public boolean contains(T s) {
    // returns whether the list contains value s
How many of the following work as the body for the clear method (empties bag)?

I. size = 0;  
   A. 0

II. head = null;  
    B. 1

III. head.next = null;  
     C. 2

IV. Node curr = head;  
    D. 3

   while(curr != null) {
     E. 4
       curr.value = null;
       curr = curr.next;
   }
How many of the following work as the body for the clear method (empties bag)?

I. size = 0;  
II. head = null;  
III. head.next = null;  
IV. Node curr = head;  
   while(curr != null) {  
      curr.value = null;  
      curr = curr.next;  
   }

A. 0  
B. 1 (II.)  
C. 2  
D. 3  
E. 4
Removal from Linked List

class List {
    private Node head;

    public boolean remove (T item) {
        // Removes the first instance of the object item
        // from the calling list, if present.
        // Returns true if item is present, false if not.
        ...  
    }
}
Suppose we have a reference (current) to the node containing the item to be removed.

What additional information do we need to successfully remove the node?
A) Nothing additional.
B) A reference to the node immediately prior to the deleted node.
C) A reference to the node immediately after the node to be deleted.
D) Both B and C.
Suppose we have a reference (current) to the node containing the item to be removed.

What additional information do we need to successfully remove the node?

A) Nothing additional.
B) A reference to the node immediately prior to the deleted node.
C) A reference to the node immediately after the node to be deleted.
D) Both B and C.