1. (2 pts) The first node in a linked list is known as the **head** of the list, and the last node in the list is known as the tail of the list and is usually indicated by having the pointer field associated with it having the value **null**.

2. (2 pts) Suppose we have a linked list whose node definitions contain an int called datum and a Node called next. Write a recursive method which accepts one parameter of type Node which will display the sum of the digits in the number stored in the Node.

   ```java
   public void traverse(Node head) {
       if (head != null) {
           int number = head.datum;
           int sum = 0;
           while (number != 0) {
               sum += number;
               number = number / 10;
           }
           System.out.println(sum);
           traverse(head.next);
       }
   }
   ```

3. (2 pts) A stack is a **LIFO** data structure and a queue is a **FIFO** data structure because of the nature of how elements are added to and removed from the structures.

4. (2 pts) The top node in a tree is called the **root**, and a node without children is called a **leaf**.

5. (2 pts) Suppose we have a LinkedList defined with one instance variable name head which is of type Node. Node is a self-referential structure that contains an instance variable of type Object called datum, and one of type Node called next. Show an implementation of an add method so that a new element is placed at the tail of the list.

   ```java
   public void add(Object datum) {
       if (head == null) {
           head = new Node(datum, head);
       } else {
           Node current = head;
           while (current.next != null) {
               current = current.next;
           }
           current.next = new Node(datum, null);
       }
   }
   ```