(a) List all of the ways a student can lose 50% or more on an assignment.

(b) Circle the bullet in front of each statement that accurately describes your instructor's course policies:

- The final exam for this course will be comprehensive.
- Lab attendance is required.
- If you miss more than three lectures, your grade will be dropped by half a letter grade.
- You must submit every lab assignment in order to pass this course.
- You will 10% extra credit by submitting the lab assignment two days before it is due.
- All students must submit a copy of their lecture notes before the end of the day.
- Having a copy of all or part of another student's source code, just to look at for a little help, is considered cheating.
- Your instructor is such a nice guy that if you turn in an assignment 45 minutes late, he wouldn't consider it late.
- Your instructor will give you a free lunch if you invite him to lunch.

Do not circle the bullet in front of any statement that incorrectly describes your instructor's course policies.
Feel free to provide an explanation for your answer if you believe the statement to be ambiguous.

Implement the following methods from the simple ArrayList class that we have been developing in lecture. You may not use any other methods from the ArrayList class in your implementation.

```
public boolean add(E obj) {

}

public boolean addAll(Collection<? extends E> collection) {
  boolean isChanged = false;

  return isChanged;
}
```

Implement the following method from the simple ArrayList class that we have been developing in lecture. You may use size() but no other methods from the ArrayList class in your implementation.

```
public boolean removeAll(Collection<? extends E> collection) {















}
```

Consider the simple LinkedList class that we have been developing in lecture. Recall that the class has two attributes:

```
  private Node head;
  private int size;
```

and that the inner Node class has two attributes:

```
  private int value;
  private Node next;
```

Suppose that the size attribute of the LinkedList class is removed. Implement the size() method for the modified class. You may not make use of any other methods in the LinkedList class.

```
public int size() {











}
```

Complete the following implementation of a PureStack.

```
public class PureStack<E> {
  private Node top;
  private int size;

  private class Node {
    private E value;
    private Node next;

    private Node(E val, Node nxt) {
      value = val;
      next = nxt;
    }
  }

  public PureStack() {
    top = null
    size = 0;
  }

  public boolean isEmpty() {
    return top==null;
  }

  public int size() {
    return size;
  }

  public E peek() {
    if(isEmpty()) {
      throw new EmptyStackException("There␣is␣nothing␣on␣the␣stack");
    }
```

Write a partial implementation of a binary tree class showing the declaration of all necessary attributes for the binary tree, the default constructor, and any inner classes (including their attributes). You do not need to include an inner class for the iterator.

Can the following tree be a red black tree (assuming you are not allowed to rebalance the tree)? If yes, identify which nodes should be red. If not, explain why not.

Why does each bucket in a hash table need to be a collection?

Suppose we had a class with a `hashCode` method that always returned -3. What would be the asymptotic time complexity for adding an object to the hash table?

Why does the API documentation for `Object.hashCode()` state: "If two objects are equal according to the equals(Object) method, then calling the hashCode method on each of the two objects must produce the same integer result."?

Why does the API documentation for `Object.hashCode()` state: "It is *not* required that if two objects are unequal according to the equals(java.lang.Object) method, then calling the hashCode method on each of the two objects must produce distinct integer results."?