1. Arrange the following Big-O expressions in order of increasing complexity. If there are any Big-O values that “match”, indicate that with an equal sign (=), e.g. \( O(\log_2 n) = O(\log_e n) \)

\( O(n \log n), O(n^2 \log n), O(n^2), O(2^n), O(\log n^2), O(2k), O(n \log n^2), O(k^2), O(2n), O(n^3) \)

Given the following partial definition for classes that implement a binary tree:

```java
class Node {  
    int value;  
    Node left, right;  
}
public class IntBinaryTree {  
    private Node root;  
    ...
}
```

2. Complete the `leafCount` method of `IntBinaryTree` so that it returns the maximum value within the tree. What is the Big-O for your code. Explain. Partial credit for pseudo-code.

```java
public int leafCount() {  
```
3. What is an invariant? (25 words or less) (For extra credit, name the Java statement most commonly associated with invariants.)


5. You are a CSC143 tutor. A student comes to you with a question about getting a JPanel to respond to keyboard events. How do you describe it to this student? (As a tutor, you cannot write the code for the student. You can describe the basic design features. You can also ask questions.)

6. In the following recursive function, indicate the base case(s) and the recursive case(s).

   ```java
   public int recurse(int a, int b) {
       System.out.println("called recurse(" + a + ", " + b + ");");
       if(b == 0) return 1;
       if(a == 1) return 0;
       return a + recurse(b - 1, a - 2);
   }
   
   Give the output generated by the following method call. Write only the output here. Use the back of page 5 for scratch paper.
   
   System.out.println("The value is " + recurse(5, 4));
   ```
7. Create an application that displays the shown window:
The window dimension is 200 x 150. There are three visible components in the window: two buttons and one label. The label displays centered, 36-pt, bold Arial text. (Adjusting the label is optional, not required.) Clicking “up” changes the value displayed at the label, increasing it by one. Clicking “dn” changes the value, decreasing it by one. For extra credit, make a click on the label cause the value to revert to 100, the initial value. No need for comments in this code. (With extra-credit, this can be implemented in 50 lines, including “import” and “}” lines.)