

Name: _____

Section: _____

Quiz #5

1. (6 points) Show (using the theorem from class) that 7^n is $o(13^n)$. [Note that this is little-o, not big-O.] You must clearly write the correct limit, clearly evaluate the limit, and then clearly state the conclusion.

2. (3 points) Assume there are methods, f , g , and h with the running times below:

- f is $O(\log n)$
- g is $O(n^3)$
- h is $O(2^n)$

What is the running time for the `foo` method below? You must state the best (smallest) running time possible in simplest terms. **Do not include any unnecessary terms in your answer.**

```
void foo() {  
    f();  
    g();  
    h();  
}
```

3. (3 points) What is the running time in terms of big-O (with respect to the variable n) of the code fragment below? You must state the best (smallest) running time possible in simplest terms.

```
for (int i = 0; i < n; i++) {  
    for (int j = 0; j < 1000000; j++) {  
        for (int k = 0; k < j; k++) {  
            System.out.println("Hi");  
        }  
    }  
}
```

4. (6 points) Circle true or false for each statement below:

- If f is $O(g)$ then f is $o(g)$. TRUE FALSE
- If f is $o(g)$ then f is $O(g)$. TRUE FALSE
- If f is $\theta(g)$ then f is $O(g)$. TRUE FALSE

5. (12 points) Assume you are working on the class below (only partially shown), which represents a general tree. Note that each Node can have an unlimited number of children. The method countNodes should return the total number of nodes in the tree, by making use of the countNodesHelper method. You must implement the countNodesHelper method so that it returns the number of nodes in the sub-tree rooted at the parameter localRoot. You must use recursion.

```
public class MyTree<T> {  
  
    private class Node {  
        T data;  
        Set<Node> children;  
    }  
  
    private Node root;  
  
    public int countNodes() {  
        return countNodesHelper(root);  
    }  
  
    private int countNodesHelper(Node localRoot) {
```