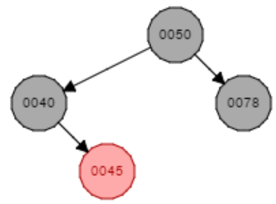
Name: ______________________________ Date: __________________

Point values are assigned for each question. Points earned: ____ / 64, ____ %

1. Show how the red-black tree would look after inserting a node with the key 0043. Use the document on Moodle that explains the insertion process succinctly. List the case you applied (i.e. 1, 2a, 3b), and write the steps you took to fix the tree (also listed in the document).



a) Draw the tree after a regular binary search tree insertion. (3 points)

b) Which property is violated? (3 points) ____
____

Case seen after regular binary search tree insertion: (3 points) _____

Steps taken to fix the tree: (3 points)

____
____

Draw the tree after taking the steps you just described. (3 points)

c) Which property is violated now? (3 points) ____
____

Case seen after first fixup: (3 points) _____

Steps taken to fix the tree: (3 points) ____
____

Draw the tree after taking the steps you just described. (3 points)

2. Draw the 2-3 tree after inserting each of the following keys. Redraw the tree for each part.
   a) 50 (1 point)

   b) 76 (1 point)

   c) 23 (3 points)

   d) 21 (3 points)

   e) 20 (3 points)

   f) 19 (3 points)

   g) 18 (3 points)

3. Read pages 241-242 in the textbook. Using that information, write pseudocode for computing the LCM of an array A[1..n] of integers. You may assume there is a working gcd() function. (6 points)

```
ALGORITHM LCM(A[1..n]):
    // Computes the least common multiple of all the integer in array A
```

4. Horner's method:

$$p(x) = 4x^4 + 5x^3 - 2x^2 - 4x + 7$$

   a. Repeatedly factor out x in the following polynomial so that you can apply Horner's method. Write your expression for $p(x)$. (5 points)

   b. Show values of the array P[0..n] as needed to apply Horner's method. (3 points)

   c. Apply Horner's method to evaluate the polynomial at $x = 2$. Make a table as we did in class showing the values x, p, n, and i, and then state your final answer for $p(2)$. (5 points)

x	**p**	**n**	**i**

   $p(2) =$ _____

   d. Use **synthetic** (not long) **division** to divide $p(x)$ by $x - 2$ to check your work. Be sure to show your work. (5 points)

5. Rewrite the *LeftRightBinaryExponentiation* algorithm on page 237 in the textbook to work for $n = 0$ as well as any positive integer. (6 points)

```
ALGORITHM LeftRightBinaryExponentiation(a, b(n)):
    // Computes a^n
```