MA/CSSE 473 – Design and Analysis of Algorithms **In-class Quiz Day 04**

Name:_______**Solution**____________ Score:____/ 9 circle your Section # 01(3rd) 02 (4th)

1. Gauss's algorithm for multiplying two complex numbers replaces _4__ integer multiplications by _3__.

2. What is the recurrence relation for the Gaussian Divide and Conquer multiplication algorithm?

   T(n) = 3 T(n/2) + Θ(n)

   What is its solution?

   In Master theorem, a = 3, b = 2, k = 1.
   So we have the third case: T(N) = $\Theta(n^{\lg 3})$,
   Which is approximately $n^{1.59}$.

3. State in your own words the (Ordinary) Principle of Mathematical induction.

   To prove that a pP(n) is true for all n> n_0, show that P(n_0) is true, and then show that for any k≥ n_0, if P(k) is true, so is P(k+1).

   Students'own words are generally fine, and should earn full credit unless they are way off.

4. Prove: For all N≥0, $\sum_{i=1}^{N} i \cdot 2^i = 2^{N+1}(N-1)+2$

   Base case: If N = 0, both sides of the equation are 0:

   Induction assumption: $\sum_{i=1}^{k} i \cdot 2^i = 2^{k+1}(k-1)+2$

   To show: $\sum_{i=1}^{k+1} i \cdot 2^i = 2^{k+2}(k)+2$

$$\sum_{i=1}^{k+1} i \cdot 2^i = 2^{k+1}(k+1) + \sum_{i=1}^{k} i \cdot 2^i = 2^{k+1}(k+1) + 2^{k+1}(k-1)+2$$

The last step uses the induction assumption. We can simply the expression on the right of the last = sign to:

$$2^{k+1}(k+1+k-1)+2 = 2^{k+1}(2k)+2 = 2^{k+2}(k)+2$$

5. Prove that any amount of postage that is 24 cents or more can be obtained using only 5-cent stamps and 7-cent stamps.
   Strong induction
   Base cases: 24 = 2*7 + 2*5,   25=5*5,   26 = 3*7 + 5,   27 = 7 + 4*5,   28 = 4* 7.

   Induction step:
   Let k be greater than 28. Asume that for any j with 24 ≤ j < k, j can be written as a linear combination of 5's and 7's. Show that k can also be.

   In particular, k – 5 can be written as 5m + 7n for some non-negative integers m and n (this is why we need five base cases, so that k-5 is not less than 24).

   Then k = j + 5 = 5m + 7 n + 5 = 5(m+1) + 7n + 5.

6. An Extended Binary Tree T with n internal nodes has __n+1__ external nodes.

7. Prove the statement from the previous question using (strong) induction, based on the definition of EBT.
   Strong induction on the number of internal nodes in T.

Let IN(T) be the # of internal nodes in the EBT T, and let EN(T) be the number of external nodes.
Base case. IN(T) = 0. This is the base case in the definition of EBT, and there is one external node.
Induction step. IN(T) > 0, so T has a root (internal node), and two subtrees TL and TR. Notice that IN(T) + IN(TL) + IN(TR) + 1, and EN(T) = EN(TL) + EN(TR).

Induction assumption. The property is true for any EBT with fewer nodes than T. In particular, for TL and TR, which are smaller because they do not include T's root. We need to use this to show that it is true for T.

EN(T) = EN(TL) + EN(TR)
= IN(TL) + 1 + IN(TR) + 1 [by the induction assumption]
= (IN(TL) + IN(TR) + 1) + 1 [rearrange and group the terms]
= IN(T) + 1

8. Tell me about anything from today's lecture that you found confusing or feel that we need to spend more time on. Be as specific as you can, (or write N/A).

   Credit as long as they write something

9. What questions do you have (from today's lecture, from the reading, or form the course in general), or write N/A.

   Credit as long as they write something