Fall 2006 **CMSC 651: Homework 2** **Due: Due Oct 4**

(This is a WRITTEN HW.)

1. (0 points) What is your name? What day is the midterm? Where is it? What day is the final? Where is it? Staple your HW.

2. (20 points) Let G be defined by

$$\begin{aligned} G(0) &= 1 \\ G(1) &= 2 \\ G(n) &= 5G(n-1) + 6G(n-2) \end{aligned}$$

   Find $G(n)$ in closed form. Show all work. (If you do not know how to solve equations of this type then look it up.)

3. (20 points) Let F be defined as

$$\begin{aligned} F(0) &= 1 \\ F(1) &= 1 \\ F(n) &= F(n-1) + F(n-2) \end{aligned}$$

   Show by a combinatorial proof (not by induction or algebra) that, for all $n, m \geq 0$,

$$F(m+n) = F(m)F(n) + F(m-1)F(n-1).$$

4. (20 points) Let U be an ordered infinite universe. Assume that comparing two element in U takes $O(1)$ steps. Design a data structure S that holds a finite subset of U such that the following holds.

   (a) INSERT can be performed on S

   (b) REMOVETOPTHIRD, which removes the top (based on the ordering) third of the elements from S, can be performed on S.

   (c) m operations can be done in $O(m)$ steps. (Prove this.)

   You may use well established algorithms as subroutines without proving their complexity.

TURN OVER- THERE IS A PAGE 2 WITH TWO MORE PROBLEMS

5. (20 points) Consider the following data structure for UNION-FIND discussed in class (called SHALLOW TREES):

   - The sets are trees of depth 1, the elements at the leaves, the name and cardinality of the set at the root.
   - FIND is easy $O(1)$- just look at your parent.
   - UNION is done by making the leaves of the smaller tree all become leaves of the bigger tree.

   We showed that n operations take $O(n \log n)$ time using an argument about counting the number of times an element was involved in a union. PROVE that n operations take $O(n \log n)$ time using a potential argument.

6. (20 points) We want to find MST for graphs where the weights are all positive integers bounded above by a number C. Give an algorithm for this that runs in time $O(CV + E)$.