CS 313	**Analysis of Algorithms**	Test 2
Name:	Summer 1999	Max Pts: 65

Important: This is an open book test. You can use any books, notes, or paper. Any calculations and rough work can be done on the back side of the test pages. You will loose five points for not writing your name.

1. [10 pt] Write a recursive function to count the number of nodes in a *circular* list.

2. [10 pt] Give asymptotic upper and lower bounds for $T(n)$ in the following recurrence. Assume that $T(n)$ is constant for $n \leq 2$. Make your bounds as tight as possible, and justify your answers.

$$T(n) = 2T(n/4) + \sqrt{n}$$

3. [10 pt] Show that any connected, undirected graph $G = (V, E)$ satisfies $|E| \geq |V| - 1$.

4. [10 pt] Show by induction that a binary tree with n nodes has height at least $\lfloor \lg n \rfloor$.

5. [10 pt] Write a recursive function that determines the location of the largest element in a binary search tree. Print the value of this largest element at the end of recursion.

6. [15 pt] Given 100 elements in an array with identical keys, what is the number of comparisons and the number of exchanges with selection sort, insertion sort, and bubble sort. Repeat the exercise with an array of 100 elements but sorted in reverse.