Name: _____

ID Number: _____

CSC 225 Midterm Exam
Oct. 21, 2009

Instructions:

1. Put your name on every page of the exam.
2. No calculators or other aids. Closed book.
3. Read through the entire exam before beginning. You should have 9 pages including this header page.

Question	Value	Mark
1	10	
2	20	
3	15	
4	15	
5	40	
Total	**100**	

Recall that you need at least 40% (40/100) in order to write the final exam in this course. Suggested strategy: read through the exam before starting, and begin with the questions which are easiest for you.

1. [10] List the data values according to the order that the cells of the binary tree are visited for each type of traversal listed.

Preorder traversal:

Postorder traversal:

2. [20] Solve the following recurrence using repeated substitution. Your answer should be a closed formula.

$T(n) = n + T(n/2), \quad T(8) = 42.$

You may assume that $n = 2^k$ for some integer $k \geq 3$.

3. [15] Prove by induction that your solution to question #2 is correct. Or for part marks [5], apply induction to the point where you realize that your solution to #1 is incorrect, and explain what goes wrong.

The recurrence from Question #1:

$T(n) = n + T(n/2), \quad T(8) = 42.$

You may assume that $n = 2^k$ for some integer $k \geq 3$.

4. Suppose that you have a heap which is stored in an array whose indexing starts at 0 (as per Java or C).

(a) [6] Give the formulas for the location of:

parent(*k*)	
leftChild(*k*)	
rightChild(*k*)	

(b) [9] Write high level pseudocode for the deleteMax operation for a heap sort routine. Your deleteMax must restore the heap after the max is deleted.

5. Suppose that Universal Studios has a large collection of old movies that they want to sort. Each movie has an integer identification number which is the key for the sort. The movies take up a lot of space (several Megabytes each) and so swapping takes an enormous amount of time; the time taken for other operations is negligible.

(a) [10] If it takes at most one hour per swap operation, then derive a tight upper bound on the time that it would take to sort *n* movies using a MaxSort, and then prove that your answer is correct using induction.

(b) [5] If you use a Heapsort, then give a function $f(n)$ such that the number of swaps in the worst case on a problem of size n is in $\Theta(f(n))$ during the deleteMax phase of the Heapsort ignoring swaps done building the initial heap.

(c) [10] Prove that the Heapsort does $\Omega(f(n))$ swaps (ignoring those done building the initial heap) for your $f(n)$ from (b). You should include in your argument a description of one example of a heap of order n such that this worst case time is realized.

(d) [10] Prove that the Heapsort does $O(f(n))$ swaps in the worst case (ignoring those done building the initial heap).

Your $f(n)$ from part (b):

(e) [5] If Universal Studios asks you which sorting routine they should use for this problem, MaxSort or Heapsort, what would your advice be and why? Justify your answer.

Use this page if you need more space.
Clearly indicate the question you are answering.