CS 146 Midterm Exam 2 **Name:_____**
Grade:

The test will be open book, open notes, 75 minute time limit. Calculators allowed, but not computers. Please write your answers on the exam sheet. Ten problems, 10 points per problem. Partial credit given only when the problem has parts (a,b,c, etc.)

1. *Analyzing recursive code using recurrence relations*

MergeSort and linked lists. We studied MergeSort in the context of sorting an array. What if we want to sort a linked list? There appears to be a problem in that it costs O(n) to find the middle of the list, while with an array that is O(1). What is the asymptotic run time of MergeSort, when implemented for a linked list using an O(n) search to find the midpoint of the list?

(a) Write a recurrence relation for the running time T(n) of this algorithm.

(b) Solve this recurrence relation to give an explicit estimate for the worst-case run time. Use O-notation.

(c) Is this faster, slower, or the same speed (up to a constant factor) as MergeSort for arrays?

2. The following algorithm is a fast way of computing x^n mod M. (It is very important in cryptography.) M will be a fixed number; we are interested in the running time for large n.

```
int fastexp(int x, int n, int M)
{ if(n==0) return 1;
  if(n==1) return x;
  int y = fastexp(x,n/2,M);
  if(n % 2 == 0) return y*y:
  return y*y*x;
}
```

For example, to compute x^8 we compute x^4 and x^2.

(a) Write a recurrence relation for the running time T(n) of this algorithm.

(b) Solve this recurrence relation to give an explicit estimate for the worst-case run time. Use O-notation.

3. *Various kinds of trees. We studied binary search trees, red-black trees, and heaps. Some of the following questions might have more than one answer.*

(a) Which of these three kinds of trees have this property? If A and B are the left and right children of some node, then we must have $A \leq B$.

(b) Which of these three kinds of trees have the property that the height of the tree is logarithmic in the number of nodes?

(c) Which of them has the property that any two paths (branches) from the root to a leaf differ in length by at most 1?

(d) Which of them has an O(lg *n*) insertion algorithm, where *n* is the number of nodes in the tree?

4. *Merge sort*. The following array is to be sorted:

19	3	6	12	1	17	33	15	8	5	22	13	10	7	9	11

(a) If *mergeSort* is called to sort this array, how many recursive calls will there be (not counting the top-level call)? ____ *(Remark: the textbook's mergeSort does get called on arrays of length 1).*

(b) What will be the maximum recursion depth, i.e., the maximum number of nested recursive calls that have not been returned from yet? ____

(c) What will the array look like after the return from the first recursive call?

5. *Quick sort.* The following array is to be sorted (it's the same as the one above):

19	3	6	12	1	17	33	15	8	5	22	13	10	7	9	11

(a) if *quickSort* is used to sort this array, what will the array look like just before the first recursive call?

(b) What will the array look like just after the first recursive call returns?

6. *Heap data structure*. Here is the array representation of a heap.

20	10	30	5	15	25	35	2	7	12	17					

Draw this heap as a tree:

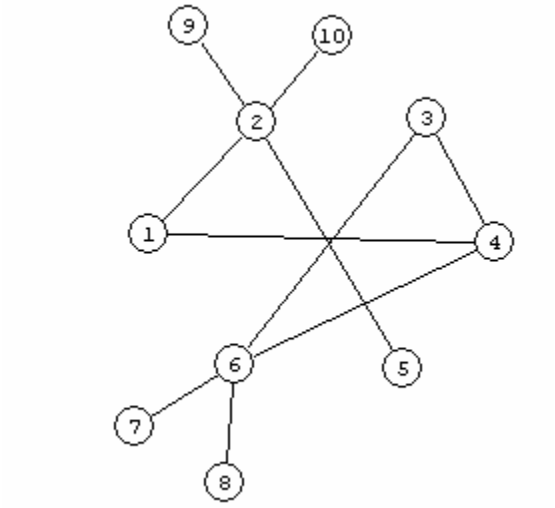
7. *Heap algorithms.* The following array is not a heap, but the heap properties are violated only once (involving two numbers).

22	15	16	14	11	13	7	10	18	2	9	8	4			

(a) Which two numbers in the array violate the heap properties?

(b) After calling MAX-HEAPIFY to correct that violation, the resulting array will be a heap. Show it in array form:

8. Consider the following graph.



If this graph is searched by **recursive** depth-first search, starting from vertex 4, and if neighbors are generated in numerical order, what will be the order in which the nodes are visited?

Answer: __, __, __, __, __, __, __, __, __, __

9. *Breadth-first search.* If the graph in the previous problem is searched by breadth-first search starting from vertex 4, and if neighbors are generated in numerical order, what will be the order in which the nodes are visited?

Answer: __,__,__,__,__,__,__,__,__,__,__,__,__,__

10. *Breadth-first search.* Consider the World Wide Web as a graph, as we did in the *Google* programming assignment. In order to gather the data that was used as the input in that assignment, namely the list of web pages and the links on that page, a "web crawler" uses breadth-first search. The time required to collect this data is obviously of interest. Assume there are N pages on the Web, with an average of L links per page.

(a) Give an estimate for the time T(N,L) needed to "crawl" the entire Web using big-O notation, counting reading a web page as one step.

Answer: _____

(b) To give a more accurate estimate we should take into consideration that it is comparatively slow to open a network connection. Assume that the time required to open a network connection requires *c* times longer than the execution of a basic step on one computer. Neglect the time it takes to download a web page once the network connection is open. Revise your estimate in part (a) accordingly, giving an expression for T(N,L,c).

Answer: _____