CS 6901 Capstone Exam Systems Spring 2012: Choose any 2 problems.

1) a) Construct a circuit diagram for a 2x4 decoder.
b) Let $F(a, b, c, d) = a'b'c'd' + a'b\,c\,d + ab'c'd + a\,b\,c\,d$. Use a 4x16 decoder (as a block diagram) and a minimal number of additional logic gates to implement F.

2) Consider the dining philosopher problem with 10 philosophers. Assume that philosopher i ($i = 0, 1, ..., 9$) executes the following:

```
while (true) {
  think;
  wait(mutex);
    wait(fork[i]);
    wait(fork[(i+1)%10];  // % is the mod operator
  signal(mutex);
  eat;
  signal(fork[i]);
  signal(fork[(i+1)%10];
}
```

All semaphores have been initialized to 1.

a) Is deadlock possible?
b) Is fairness guaranteed? That is, is indefinite postponement for an individual philosopher possible?
c) Describe any other undesirable aspects of this proposed solution, if any.

Explain your answers.

3) Consider a system with 4 resources (A, B, C, D) in quantity (5, 3, 3, 3). The Banker's Algorithm is used to allocate resources and it has the following SAFE state:

Available: A B C D
1 2 1 0

Process	Allocation				Max				Need			
	A	B	C	D	A	B	C	D	A	B	C	D
P0	3	0	1	2	5	3	3	3	2	3	2	1
P1	1	1	1	0	2	3	2	1	1	2	1	1
P2	0	0	0	1	0	1	1	1	0	1	1	0

a) Justify why the current state is safe.
b) Will a request for P0 of (1, 1, 0, 0) be allowed? Justify your answer.

CS 6901 Capstone Exam Data Structures Spring 2012: Choose any 2 problems.

1. Write an efficient function to sort the elements of a linked list of integer values. Your sort must be in-place; you may not dynamically allocate new memory. Code in the language of your choice and include declarations of all data structures.

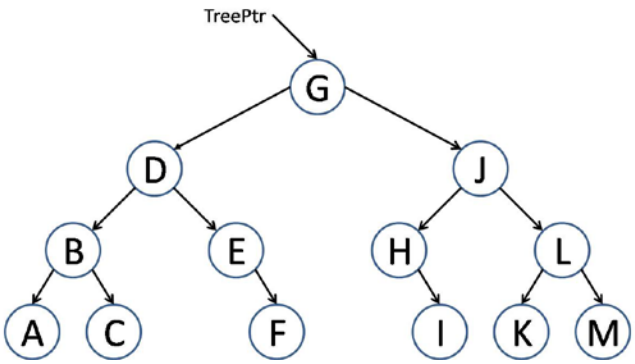
   Example:

   Input: listptr $\rightarrow 11 \rightarrow 8 \rightarrow 2 \rightarrow 4 \rightarrow 5$

   Output: listptr $\rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 8 \rightarrow 11$

2. Given a possibly empty binary search tree containing character values, write a recursive routine that takes as input an integer *D* and produces as output an ordered list of all characters at depth *D* in the tree. Code in the language of your choice and include declarations of all data structures.

   Example:



   For *D* = 2, your routine should display B E H L
   For *D* = 3, your routine should display A C F I K M

3. Solve the recurrence relation $T(n) = T(n/2) + \lg(n)$ where $T(1) = 1$ and $n = 2^k$ for a nonnegative integer *k*. Your answer should be a precise function of n in closed form. (An asymptotic answer is not acceptable.) Show the work you did to obtain the solution. Note that lg represents the base 2 log function.

CS 6901 Capstone Exam Theory Spring 2012: Choose any 2 problems.

1. Prove that $\{0^a1^b0^c : b \neq a + c; a, b, c \geq 0\}$ is a context-free language.

2. Consider the following two languages:

   $HALT_{TM} = \{M, w : M \text{ is a Turing machine that halts on input string } w\}$
   $TWO_{TM} = \{M : M \text{ is a Turing machine that accepts exactly two strings}\}$

   $HALT_{TM}$ (the Halting Problem) is, of course, undecidable. Through reducibility from $HALT_{TM}$, show that TWO_{TM} is also undecidable.

3. In formal logic, a **tautology** is a Boolean formula that always evaluates true. Prove that the language

   $NOTAUT = \{\varphi : \varphi \text{ is a Boolean formula that is } \textbf{\textit{not}} \text{ a tautology}\}$

   is in the class NP.