CS 6901 Capstone Exam Systems Winter 2017: Choose any 2 of the 3 problems.

1) Design a fully simplified combinational circuit as follows:
3 inputs: $a_2a_1a_0$ treated as a 3-bit signed integer a
3 outputs: $b_2b_1b_0$ to be the 2's complement of a
Show your work and draw the resulting circuit diagram.

2) There are 3 standard goals to the 2-process mutual exclusion problem.
Goal 1: Mutual exclusion is guaranteed
Goal 2: Deadlock cannot occur.
Goal 3: Indefinite postponement cannot occur.

Attempted Solution: common variables: flag1, flag2 (both initially false)

```
   Process 1
while (true) {
  flag1 = true;
  while (flag2); //empty body
  Critical section;
  flag1 = false;
  Noncritical section;
}
```

```
   Process 2
while (true) {
  flag2 = true;
  while (flag1); //empty body
  Critical section;
  flag2 = false;
  Noncritical section;
```

For the above solution,
a) Select one goal that is not satisfied and provide an execution sequence that violates the goal.
b) Select one goal that is satisfied and give a brief explanation that justifies why the goal is met for all possible execution sequences.

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	0	1

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

a) Justify why the current state is safe.
b) If P0 requests an additional unit of resource B, will it be allowed? Justify your answer.

CS 6901 Capstone Exam Data Structures and Algorithms Winter 2017

Choose any 2 of the 3 problems.

1) Given a (possibly empty) binary search tree of integers, write a function that constructs a singly linked list of the tree’s entries in ascending order. Return a pointer to the first entry in the list.

2) Implement a FIFO queue of integers using a circular array a[0] .. a[n-1], where n is a constant.

a) Declare the data structure.

b) Write a function that is given the circular array queue and an integer x and returns true if and only if x is an entry in the queue.

3) For each function with input argument n, determine the asymptotic number of “fundamental operations” that will be executed. Note that `fc` is recursive. Choose each answer from among the following. You do not need to explain your choices.

$\theta(1)$ $\theta(\log n)$ $\theta(n)$ $\theta(n \log n)$ $\theta(n^2)$ $\theta(n^2 \log n)$ $\theta(n^3)$ $\theta(2^n)$ $\theta(n!)$

a)
```
void fa(int n) {
  for(k = 1; k < n; k++) {
    Perform 1 fundamental operation;
    for(i = k-1; i >= 0; i--)
      Perform 1 fundamental operation;
    //endfor i
    Perform 1 fundamental operation;
  }//endfor k
}
```

b)
```
void fb(int n) {
  for(i = n/2; i <= n; i++)
    Perform 1 fundamental operation;
  //endfor i
}
```

c)
```
void fc(int n) {
  if (n == 1)
    Perform 1 fundamental operation;
  else {
    fc(n-1);
    Perform 1 fundamental operation;
    fc(n-1);
  }//endif
}
```

Theory Exam

Answer **ANY TWO** of the following three questions:

1. Consider these problems. Note that {} is the empty set ∅

   Given a DFA **D**, does **D** accept at least one string? That is, is L(**D**) ≠ {}?

   Given a context free grammar **G,** does **G** generate at least one string? That is, is L(**G**) ≠ {}?

   Given a Turing machine **M,** does **M a**ccept at least one string? That is, is L(**M**) ≠ {}?

   a. For each problem, is it decidable or not?

   b. Choose one of the three problems, and prove that it is (or is not) decidable.

2. Prove KNAPSACK is NP-complete. You may assume that CNFSAT, 3-CNF-SAT, VERTEX_COVER, HAMILTONIAN_CIRCUIT, 3-DIMENSIONAL_MATCHING and SUBSET_SUM are all known to be NP-complete.

[If you need a reminder of the statements of these problems, a sheet is available with their statements.]

KNAPSACK: Given a finite set of n items **S** = {i_1 , i_2, … i_n }, where each item has a positive integer weight (w_i) and positive integer value (v_i), with positive numbers B and T, does **S** have a subset **S'** such that:

(1) the sum of the weights in **S'** is less than or equal to B.

and

(2) the sum of the values in **S'** is greater than or equal to T?

Example yes-instance with 5 items; B=6, T=18

Item	I_1	I_2	I_3	I_4	I_5
Weight	1	3	1	2	2
Value	5	7	1	7	3

3. a. Write a context free grammar (or Backus Naur form grammar) for strings of identifiers with two binary operators, $ and & where:

   Terminal symbols: {id, (,), $, &}

   $ and & are both left associative

   & has higher precedence than $

   (and) have the usual meaning that operations inside () are done first

   Example string: id & (id $ id & id)

   b. Write a regular expression for L = {w | w in {a,b,c}* such that w contains the substring "cc" ***or*** w contains an odd number of a's (or both). The strings in L may have additional c's, and may have any number of b's}