## ENEE 350 TEST 1-Spring 2005 CLOSED BOOK AND NOTES EXAM PERIOD 75 MINUTES

## Instructions:

- Points for each problem are indicated right after the problem. The total score is 100 points.
- Use the space provided below each problem. if you need more space, please ask a proctor.
- Write your name and student id on the cover sheet.
- Promptly hand in your test to a proctor when the test is over.

NAME:
STUDENT ID:

**Problem 1 (20 points):** Consider the following vesp program. Specify the values in memory locations 0, 1, and 10 after the execution of the program is completed, and explain why.

100	LDA	0, 50
102	LDA	1,-15
104	ADD;	
105	MOV	10,0;
107	JPS	104;
108	HLT;	

Note: All addresses and operand values are expressed in decimal.

## Problem 2 (20 points):

The computation engine of a given processor executes four logic instructions on a pair of 4-bit operands,

$$A = [a_0,a_1,a_2,a_3]$$

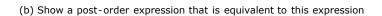
$$B = [b_0,b_1,b_2,b_3]$$

whose instruction format is given below:

Instruction	Op-code	Operation
Сору В	00	A = B
Complement A	01	A = ~A
Left shift A	10	A = A << 1
Right shift A	11	A = A >> 1

Design a combinational circuit that can carry out these four instructions in the computation engine of this processor. (*Note: You are only required to describe a combinational circuit that performs these logic instructions, not the entire computation engine of the processor.*)

Problem 3 (20 points): Consider the following expression			
	a *(4+ b *( 8+ c *(16 + d*(e + f))))		
(a) Draw a hinary tree that represents this e	expression		



- (c) Show a pre-order expression that is equivalent to this expression
- (d) Write a CodeMill stack program for the pre-order expression you obtained in part (c)

(e) Clearly show the first 5 frames of CodeMill's stack during the execution of your program assuming that the stack pointer is initially set to the floor of the stack.

**Problem 4 (20 points):** The following code describes the instruction sequencer of a central processor unit. Give a control circuit that implements this instruction sequencer, clearly indicating the events that occur in each state.

```
while(reset == 0)
{
Step 1: X = Y; Y = Y +4;
Step 2: U = MEMORY[X];
Step 3: switch(U[0:2])
{
  case 0: A = A +1;    break;    case 1: A = A - 1;    break;
  case 2: A = A + B;    break;    case 3: A = A - B;    break;
  case 4: A = 2A;         break;    case 5: A = A/2;         break;
  case 6: A = 0;         break;    case 7: A = 1;         break;
}
```

**Problem 5 (20 points):** Write a vesp program to sort the numbers in memory locations 120,121,122 in ascending order. For example, if the numbers in locations 120, 121, and 122 are 4, -2 17, after sorting, they will be -2, 4, 17.