Quiz 2 CSCI 255 Spring 2001 9 April, 2001

Name: ____

This is a closed book exam. Use of calculators is also not allowed. Be sure to show your work in order to get full credit for the problem. When possible place your answers in the provided boxes. There are 9 questions for a total of 150 points on this quiz.

Problem 1 (64 points):

3:15-3:47

In this problem you are asked to write **eight** *independent* sections of LC-2 assembly code to set registers R0 or R1 or LC-2 memory locations based on constants, the present values of R3 and R4, or LC-2 memory locations. You may use registers R6 or R7 as "scratch" registers but should not modify any other registers. You must assume that your code will be located somewhere between memory locations x3000 and x30FF. You may use .fill's when needed to initial memory locations. You should assume that these .fill's would also be stored in memory locations x3000 to x30FF.

In these subproblems, the code to implement is given in the psuedo-C notation used in class lectures. Rn will be used as a reference to LC-2 register n. M[n] will be used as a reference to LC-2 memory location n.

```
R0 \leftarrow 5 * R3 ;
R0 \leftarrow R3 - R4 ;
R0 \leftarrow R3 \& R4 ;
if (R3 == 15) \\ R0 \leftarrow R4 ;
else \\ R0 \leftarrow R4 + 1 ;
R0 \leftarrow R4 + 1 ;
R0 \leftarrow R4 ;
while (R0 < 107) \\ R0 \leftarrow R0 + R0 ;
```

 $M[x3100] \leftarrow M[x3100] + 5;$

 $M[x4100] \leftarrow M[x4100] + 5;$

 $R0 \leftarrow R3 + 1 ;$ $R1 \leftarrow R4 + '1' ;$

Problem 2 (16 points):

3:47-3:55

Translate into LC-2 machine language (binary) program the LC-2 assembly language program shown below:

Pro8		
	.ORIG	x3000
	LD	R1,MX
	LDI	R2,MX
	LEA	R3,MX
	LDR	R4,R1,#1
	HALT	
MX	.FILL	0x3006
MY	.FILL	0x3007
ΜZ	.FILL	0x3008
	.END	

Problem 3 (12 points):

3:55-4:01

What are the values of registers R1 to R4 after the LC-2 assembly language program in Problem 2 is executed?

R1	=	
R2	=	
R3	=	
R4	=	

Problem 4 (8 points):

What Linux command would you use to assemble the LC-2 assembly program lab9.asm? Give not only the name of the command, but the arguments you use with it.

Problem 5 (10 points):

Write some LC-2 assembly code to write the contents of register 5 to the CRT using the CRT data and status registers?

Problem 6 (8 points):

4:10-4:14 Write some LC-2 assembly code to write the contents of register 5 to the CRT using a LC-2 trap routine?

Problem 7 (12 points):

The VAX computer has an instruction called BIC (Bit Clear) that performs the logical operation $\alpha \beta'$. Write a LC-2 subroutine called BIC in assembly language that performs this operation on registers R0 and R1, that is, the subroutine performs the operation:

 $R0 \leftarrow R0 \& \sim R1 ;$

Problem 8 (8 points):

Show the complete LC-2 instruction needed to call the BIC subroutine of Problem 7. You may assume that both the calling and called subroutine are on the same page.

Problem 9 (10 points):

Translate the following two LC-2 binary instructions into LC-2 assembly code.

0001011011111110	
0101011011000111	

4:14-4:20

4:01-4:05

4:05-4:10

4:20-4:24

4:24-4:30

LC-2 traps		
Number	Name	
\$20	GETC	
\$21	OUT	
\$22	PUTS	
\$23	IN	
\$24	PUTS	
\$25	HALT	

Quiz 2 reference and tally page	

LC-2 hardware registers		
Location	Name	
\$F3FC	CRT Status Register	
\$F3FF	CRT Data Register	
\$F400	Keyboard Status Register	
\$F401	Keyboard Data Register	

LC-2 instruction format

1	5 14 15 1	e 11 10 9	076	5 4 5 2 1
ADD*	0001	DR	SR1	0 00 SR2
ADD*	0001	DR	SR1	1 imm5
AND	0101	DR	SR1	0 00 SR2
ANT [0101	DR	SR1	1 imm5
BR [0000	n z p		Pgoffsat9
JSR [0100	L 00	- i . 	pgoffsat9
ISRR	1100	L w	BaseR	index6
	0010	DR.		pgoffsat9
LDI* [1010	DR.		pgoffsat9
LDR+	0110	DR.	BaseR	index6
	1110	DR		pgotfset9
иот•	1001	DR	SR	111111
RET	1101		00000	000000
рт т '	1000	111	00000	
sт [0011	SR		Pgoffsat9
sm [1011	SR		P@offset9
STR [0111	SR	BaseR	index6
TRAP				trapvedt8

1	/64
2	/16
3	/12
4	/8
5	/10
6	/8
7	/12
8	/8
9	/12
	/150