

**CS 2734, Computer Organization II**  
**Fall Semester, 2000**  
*First Examination*

1. Below are questions about number representations and conversions: (20)

(a) Show how to represent the number  $-82$  as a 32-bit two's complement binary number. (82 in binary is 1010010.)

(b) Consider the floating point number (a double) with hexadecimal and binary representations:

c0240000 00000000 (hexademcinal)  
 1100 0000 0010 0100 (48 more 0's) (binary)

What is the number? (Show your work. Remember that the bias for a double is 1023, and that an exponent of 1 is represented by 100 0000 0000.)

2. Consider the following MIPS code fragment: (20)

```
.data
A:   .word 47, 23, 89, 52, 43, 0, 0, 0, 0, 0,
      .text
# insert MIPS instructions here.
```

For insertion at the comment, write MIPS instructions that will do the following:

- (a) Store the address of A into  $\$s0$ .
- (b) Use a loop to print the first 5 values of the array A, each separated by a blank, so that 47 23 89 52 43 should be printed.  
 (You must use a loop for credit. If you like, you may take advantage of the fact that the first five values are followed by 0's. Your MIPS code should do what is asked for above and *nothing more*.)

3. Write a MIPS function F so that (20)

- (a) F saves register  $\$ra$  *on the stack*.
- (b) F doubles its input parameter and returns the doubled value.
- (c) F restores the register  $\$ra$  saved above and returns.
- (d) Separately show a call to F with input parameter 19.

Note: You should just give code for the call to F and for the definition of the function F that do the above items and *nothing more*. You should follow MIPS parameter conventions.

(20)

4. Consider the following assembler instruction:

```
beq $t0, $t1, Label
```

- Give the machine code for this instruction. You can simply give the decimal value of the bits in each of the fields of the instruction, so you do not need to convert to binary or hexadecimal. Note that register `$t0` is 8 and register `$t1` is 9. Assume that the address of `Label` is 24 bytes beyond the address of this instruction.
- This instruction will not work in case `Label` is “too far away.” Say precisely how far “too far away” is. (Be sure to say whether your answer is in bytes or words.)
- In case `Label` is too far away as in (b) above, show how the `beq` instruction could be changed to two instructions that would always work.

5. Consider the following logic gate constructed out of CMOS transistors.

(20)

- In case `A` is a 1 (voltage high) and `B` is a 0, what will be the value at `C` and the output at `D`? Explain your answer in terms of the diagram and the properties of the transistors. (Show which switches are open, which are closed.)
- What kind of gate does this diagram represent? (Explain.)

