CSCE 212: Exam 1 **Fall 2009**

Name (please print):_____ **Total points: ___/60**

Instructions
This is a CLOSED BOOK and CLOSED NOTES quiz. However, you may use calculators, scratch paper, and the green MIPS reference card from your textbook. Ask the instructor if you have any questions. Good luck!

1. (20 points) Write a snippet of MIPS code that reverses the bit sequence stored in register $s0. You may use the following algorithm (substitute symbolic registers with actual registers):

   *1) set register **A** to 0 (used as counter)*
   *2) set register **B** to 0 (used as result)*
   *3) right-shift $s0 by **A** bits, put result into temporary register **C***
   *4) left-shift **C** by 31 bits*
   *5) right-shift **C** by A bits*
   *6) perform a bit-wise OR between register **B** and **C**, put result in **B***
   *7) increment **A***
   *8) if **A** < 32, go to line 3*

2. (10 points) Assume there's a pseudo-instruction named **`die`**, which causes the machine to lock up by creating an infinite loop. Show how this instruction could be translated (using assembly code).

3. (20 points) Write a MIPS subroutine called **div8** that returns 1 in register $v0 if the argument in register $a0 is evenly divisible by 8 (and 0 otherwise). *Hint: the check can be performed using a shift instruction and a **beqz** or **bnez** instruction.*

   The subroutine must not change any of the caller's registers (except for $v0), meaning that it must use the stack to save the original values of any registers that it uses.

4. (10 points) Convert the following sequence of machine instructions, represented as hexadecimal values, into assembly language.

   ```
   3c010010
   34300001
   01105006
   ```