NAME: ______________________________________ STUDENT #: ______________________ L16-1

EECE 259: Introduction to Microcomputers **Lecture Quiz** **Feb 8, 2011**

1. The Fibonacci numbers are $F_{i+1} = F_i + F_{i-1}$ where $F_1=1$ and $F_0=0$. Write:
   a. Main program and location FIB to call genFibTable(FIB,N), lookupFib(FIB,13) and findFibLE(FIB,180).
   b. Subroutine **lookupFib(*X*,*Y*)**. Given a table of Fibonacci numbers starting at address *X* in *r4* and a value *Y* in *r5*, where $0 \leq Y < N$, find F_Y in the table and return it in register *r2*.
   c. Subroutine **findFibLE(*X*,*Z*)**. Given a table of Fibonacci numbers starting at address *X* in *r4*, and a value *Z* in *r5*, where $0 \leq Z < F_{N-1}$, find the largest Fibonacci number $F_i \leq Z$ and return it in register *r2*.
   d. Subroutine **genFibTable(*X*,*Y*)**. Compute the first *Y* Fibonacci numbers and store them *in ascending order* in memory. Start the table at location *X* in register *r4*, and value *Y* is in register *r5*.

```
.equ    N, 64
.global _start:
.text
/* ********************************************************************** */
_start:         movia   r4, FIB
                movi    r5, N
                call    genFibTable
                movia   r4, FIB
                movi    r5, 13
                call    lookupFib
                movia   r4, FIB
                movi    r5, 180
                call    findFibLE
STOP:           br      STOP

/* ********************************************************************** */
lookupFib:      muli    r2, r5, 4
                add     r2, r2, r4  /* desired table address = r4 + 4*r5 */
                ldw     r2, 0(r2)
                ret

/* ********************************************************************** */
findFibLE:      ldw     r2, 0(r4)       /* r2 holds F(i) value */
                addi    r4, r4, 4       /* next table entry address */
                ble     r2, r5, findFibLE /* if F(i) <= Z, keep searching */
                ldw     r2, -8(r4)      /* F(i) > Z, so load F(i-1) in r2 */
                ret

/* ********************************************************************** */
genFibTable:    stw     r0, 0(r4)      /* store F(0) = 0 */
                movi    r2, 1
                stw     r2, 4(r4)      /* store F(1) = 1 */

                movi    r6, 2          /* # of Fibs computed so far */

genLoop:        ldw     r2, 0(r4)      /* read F(i-1) */
                ldw     r3, 4(r4)      /* read F(i) */
                add     r2, r2, r3     /* compute F(i+1) = F(i)+F(i-1) */
                stw     r2, 8(r4)      /* store F(i+1) */

                addi    r4, r4, 4      /* next table entry address */
                addi    r6, r6, 1      /* increment # of Fibs computed */
                blt     r6, r5, genLoop

                ret

.data
FIB:
.skip   4*N
.end
```