**2.31** [5] <§2.8> Implement the following C code in MIPS assembly. What is the

total number of MIPS instructions needed to execute the function?

int fib(int n){

if (n==0)

return 0;

else if (n == 1)

return 1;

else

return fib(n−1) + fib(n−2);

**2.34** Translate function f into MIPS assembly language. If you need to use

registers $t0 through $t7, use the lower-numbered registers fi rst. Assume the

function declaration for func is “int f(int a, int b);”. Th e code for function

f is as follows:

int f(int a, int b, int c, int d){

return func(func(a,b),c+d);

}

1. Solve the following 9-bit 2’s complement binary subtractions:

• 144 - 28 = ?

• 92 - 43 = ?

• -32 - 84 = ?

For each of these subtractions, convert the numbers into 9-bit 2’s complement, add them together, and convert the resultant binary 2’s complement number into decimal form. You are expected to show all your work.

**3.12** [20] <§3.3> Using a table similar to that shown in Figure 3.6, calculate the

product of the octal unsigned 6-bit integers 62 and 12 using the hardware described

in Figure 3.3. You should show the contents of each register on each step.

**3.13** [20] <§3.3> Using a table similar to that shown in Figure 3.6, calculate the

product of the hexadecimal unsigned 8-bit integers 62 and 12 using the hardware

described in Figure 3.5. You should show the contents of each register on each step.

**3.18** [20] <§3.4> Using a table similar to that shown in Figure 3.10, calculate

74 divided by 21 using the hardware described in Figure 3.8. You should show

the contents of each register on each step. Assume both inputs are unsigned 6-bit

integers.

**3.20** [5] <§3.5> What decimal number does the bit pattern 0×0C000000

represent if it is a two’s complement integer? An unsigned integer?

**3.22** [10] <§3.5> What decimal number does the bit pattern 0×0C000000

represent if it is a fl oating point number? Use the IEEE 754 standard.

**3.23** [10] <§3.5> Write down the binary representation of the decimal number

63.25 assuming the IEEE 754 single precision format.

**3.24** [10] <§3.5> Write down the binary representation of the decimal number

63.25 assuming the IEEE 754 double precision format.