Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Computer Organization (CS 345)**

**Exam 2**

Part 1: Short Answer (30 points)

a) Assembly (4 points each). Write MIPS assembly code for each of the following:

1. Allocate 3 double variables in the .data section called x, y, and z, setting their values to 2.0, 4.0, and 6.0.

1. Load x, y, and z into registers $f0, $f2, and $f4, respectively.
2. Add x + y + z and store the result in the register representing z.
3. Store the result in the memory location for z.

b) Boolean algebra (2 points each) reduce each of the following Boolean expressions. Assume A, B, C, and D are Boolean variables. ~ = NOT, \* = AND, and + = OR. 1 = true and 0 = false.

1. A\*0 + B\*0
2. (A + 1) \* (B + 1)
3. (A + B) \* C + D
4. D \* (A + B + C + 1)

d) Bits and bytes (2 points each)

1. How many bytes are in a 32-bit integer?
2. How many bits are in a byte?
3. How many bytes are used to represent one ASCII character?

Part 2: Long Answer (70 points, 10 points each)

1. Convert the following C/Java statements to MIPS assembly code using $t0 for x and $t1 for i, respectively:

int x = 0;

for(int i = 0; i < 10; i++)

 x = x + i;

2. Convert the following MIPS statement from assembly to a 32 bit binary number and then to hex. The hex value of the opcode for addi is 0x8. The decimal value for $t3 is 11d. The decimal value for $s2 is 18d.

 addi $t3, $s2, 53

3. Convert the following numbers to two’s complement using 8 bits. Remember that one bit will represent the sign. Then add the two numbers.

 24d

 -31d

4. Assume you are given a recursive function called Factorial that computes the factorial of any non-negative integer. The factorial of a number n (also written n!) is n\*n-1\*…\*2\*1. The factorial of zero, 0! = 1, and the factorial of one, 1! = 1. So, the factorial of 2, 2! = 2\*1. Assume that there is only one parameter to the factorial function. Perform a stack trace of Factorial(2). You must show how the stack pointer is adjusted after each function call. Hint: be sure to keep track of the return address and the current value of the parameter.

5. Convert the following decimal value to binary:

304.875d

Put the binary number above in scientific notation. That is, convert it to a value in the format: x.xxxxxxxxx \* 2^y. Determine the value of y in both decimal and binary.

6. Write a function in MIPS to compute the cube of a double variable. The cube of x is x^3 or x\*x\*x. You may assume the double value is stored in a memory address that will be passed in as the parameter $a0. You may assume the result will be stored in memory at an address provided in the parameter $a1. Hint: be sure to load the value into a double register using l.d and store the result in memory using s.d. This function should not return any values.

7. Explain the fetch-decode-execute cycle that is carried out by the CPU. Be sure to describe the role of the Control Unit, Program Counter, Registers, ALU, and Memory.