Name__________________________________

CS 230

Practice Midterm

Instructions: closed books, closed notes, open minds,
1 hour 15 minute time limit.
There are 5 sections for a total of 75 points:

Part I:	**16 points**
Part II:	**12 points**
Part III:	**25 points**
Part IV:	**12 points**
Part V:	**9 points**

Preserved	Not preserved
Saved registers: \$s0 - \$s7	Temporary registers: \$t0 - \$t9
Stack pointer register: \$sp	Argument registers: \$a0 - \$a3
Return address register: \$ra	
Stack above the stack pointer	Stack below the stack pointer

CPU time (sec) = Instruction count x CPI (cycles/instr..) x Clock cycle time (sec/cycle)

$$\text{CPU time (sec)} = \frac{\text{Instruction count x CPI (cycles/instr.)}}{\text{Clock rate (cycles/sec.)}}$$

1 MHz = 10^6 cycles/sec

Part I. Define or describe the following: (10 points total)
(actual exam will ask you to match terms with descriptions)

1. Hardware Level: Integrated Circuit, Transistor, Switch, Gate

2. Clock

3. Basic Components: Central Processing Unit (CPU), ALU, Register

4. Operating System

5. Program translation: Assembler, Compiler

6. Performance testing terms: Throughput, Benchmark Suite

7. Data Representation: Binary, Octal, Decimal, Hexadecimal, Floating Point, ASCII, Unicode

8. Abstraction

9. Instruction (as in assembly language instruction)

Part II. Numbers (3 points each)

10. Write the four-bit (including sign bit) two's complement representation for the numbers -5 and 6.

11. Show how to carry out the operation (-5) - 6 in four-bit two's-complement notation.

12. How can you tell whether an overflow occurs in a finite-precision two's-complement addition or subtraction? Did an overflow occur in the operation in question 11?

13. The hexadecimal number found in a single precision floating point register is 0xC1660000. Assuming the IEEE 754 standard representation for single precision floating point numbers is used, what number, in decimal, does this represent?

Part III. Assembly code

14. Write MIPS assembly code that will execute the following C statement for integers `a, b, c, x`. Assume that registers are used to store variables as follows (do not worry about variables in memory): `a` in $s0, `b` in $s1, `c` in $s2, `x` in $s3. Include comments. **Do not use pseudo-instructions.** (3 points)

```
x = (a + b) - c;
```

15. Write MIPS assembly code for the following loop. Assume that register $t0 is used for variable `k`, register $s0 is used for variable `num`, and register $s1 is used for variable `sum`. Include comments. **Do not use pseudo-instructions.** (7 points)

```
sum = 0;
for(k = 0; k < num; k++)
{
     sum += k;
}
```

16. Write MIPS assembly code for the following C function. Assume that register \$a0 is used for the location of the first element of `A` in memory, that `A` contains exactly 32 elements, that register \$v0 is used for the return value, and that register \$ra is used for the return address. Include comments. **Do not use pseudo-instructions.** (Note: $32 = 2^5$.) (15 points)

```
int avg (int A[])
  // precondition: A has 32 elements
{
     int k;
     int sum = 0;
     for(k = 0; k < 32; k++)
     {
          sum += A[k];
     }
     return sum / 32;
}
```

Part IV. Truth Tables, Circuits and Gates

17. Show the truth table for a two-input *exclusive-or* function and implement it using AND, OR, and NOT gates. (Reminder: The output of the exclusive-or function is true if and only if exactly one of its inputs is true.) (7 points)

18. Two important theorems of Boolean logic are DeMorgan's Laws:

`(A OR B)' = A' AND B'` where `A'` means `NOT A`

`(A AND B)' = A' OR B'`

Fill in the following truth table and explain how it proves that DeMorgan's Laws are true. (5 points)

A	B	A'	B'	(A OR B)'	A' AND B'	(A AND B)'	A' OR B'

Part V. Suppose you have the following instruction set mix: (3 points each)

Instr type	CPI	% in program
A	3	60
B	5	30
C	8	10

19. What is the average CPI for this mix?

20. If the clock rate is 70 GHz, how many instructions per second are executed?

21. If a program with this instruction mix executes 2 billion instructions when run on a certain data set, how long does it take to run?