CS 351 Exam 1 Mon. 9/28/2015

Name: ________________________________

Rules and Hints

- You may use one handwritten 8.5 × 11" cheat sheet (front and back). This is the only additional resource you may consult during this exam. *No calculators.*
- You may write your answers in the form *[mathematical expression][units]*. There is no need to actually do the arithmetic.
- You may use extra scratch paper if you need more space, but make it clear where to find your answer to each question.

Grade

	Your Score	**Max Score**
Problem 1: Short answer		10
Problem 2: Processor performance and power		35
Problem 3: MIPS functions		30
Problem 4: MIPS arrays		25
Total		**100**

Problem 1: Short answer (10 points)

Part A (2 points)

+1 extra credit point if entire class gets it right.

How many bits are in a byte?

Part B (4 points)

A manufacturer is considering 2 different options for the next version of their quad-core processor. In Option A, they will double the number of cores but keep the maximum per-core clock speed the same. In Option B, they will raise the maximum per-core clock speed by 0.5 GHz but keep the number of cores the same.

Give one argument in favor of Option A, and one argument in favor of Option B.

Part C (4 points)

A snippet of MIPS code (shown here in assembly) is stored in memory at address 0xaf30:

```
ADD $v0, $v0, $s0
JR $ra
```

If the contents of the register $t0 are currently 0xaf34, what will be the contents of $t1 after the following instruction is executed? Give your answer in binary or hex.

```
LBU $t1, -4($t0)
```

Problem 2: Processor performance and power (35 points)

You have a workload with 10 billion dynamic instructions. 50% are integer instructions, 20% are branches, and 30% are memory instructions.

You want to run it on a processor with the following characteristics:

- 6 cores, each with a maximum frequency of 2 GHz
- Average CPI of 1 for integer instructions; 2 for branches; 4 for memory
- Peak power consumption of 60 W (20 W static, 40 W dynamic)

Part A: Execution time (10 points)

What is the execution time of your workload if it runs single-threaded on a single core of this processor?

Part B: Optimization (12 points)

You run this program on a processor that is identical to the original version, except that the clock frequency is increased to 2.5 GHz. What will be the speedup, compared to the old version?

You recompile your source with all compiler optimizations enabled, which eliminates half of the integer instructions. What will be the speedup of your new executable, compared to the original executable?

Part C: Power and energy (13 points)

If the 2.5 GHz version of your processor also requires the operating voltage to increase by 30%, what will be the total power consumption of this new processor?

How much energy will it use to run your program?

Problem 3: MIPS functions (30 points)

Translate the following C functions to MIPS. You should translate each function independently and not attempt to combine them. Obey all MIPS conventions about functions, registers, and stack usage. You can use ASCII character values as immediate operands (so it's OK to use `'A'` as a constant).

```
int f(int x, int y) {
   // rand returns an int.
   if (rand() > x) return y + 2;
   return x + 1;
}

char g(char* x, char* y) {
   // strpbrk returns a char*
   char* a = strpbrk(y, x);

   if (!a) return '0';

   // isdigit returns a 1 or a 0
   if (isdigit(a[0])) return '1';

   return x[0];
}
```

Problem 4: MIPS arrays (25 points)

Assume that A is an array of 200 integers, starting at the top of the stack. Write MIPS code to set $s1 equal to the sum of the even integers in the array. In other words, if the array contents were [1, 5, 6, 2, 3], then the final value in $s1 would be 8 ($= 6 + 2$). Use temporary registers for all intermediate values.