The University of Alabama in Huntsville
Electrical & Computer Engineering Department
CPE 431 01
Final Exam
December 8, 2015

Name: ________________________________

1. (1 point) ________________________ multithreading is a version of hardware multithreading that implies switching between threads after every instruction.
2. (1 point) A ________________ is a lightweight process, groups of them commonly share a single address space.
3. (1 point) A _______ is a synchronization device that allows only one processor to access data at a time.
4. (1 point) _________________ is an API for shared memory multiprocessing that includes compiler directives, a library and runtime directives.
5. (1 point) ___________ usually don't have caches.
6. (7 points) What number does 0x3FE9 A700 0000 0000 0000 0000 0000 0000 represent, assuming the IEE 754 double precision format?

7. (15 points) Here is a series of address references given as hexadecimal word addresses: 1, 4, 8, 5, 20, 17, 19, 56, 209, 11, 4, 43, 5, 36, 8, 16, 59, 187. Assuming a direct mapped cache with eight word blocks, a total size of 32 words that is initially empty, (a) label each reference in the list as a hit or a miss and (b) show the entire history of the cache, including tag and data.

Word Address (Hexadecimal)	Word Address (Decimal)	Binary	Miss/Hit
0x1	1		
0x4	4		
0x8	8		
0x5	5		
0x20	32		
0x17	23		
0x19	25		
0x56	86		
0x209	521		
0x11	17		
0x4	4		
0x43	67		
0x5	5		
0x36	54		
0x8	8		
0x16	22		
0x59	89		
0x187	295		

8. (10 points) Consider the following portions of three programs running at the same time on three processors in a symmetric multicore processor (SMP). Assume that before this code is run, w is 2, x is 4 and y is 3 and z is 1. w, x, y, and z are type `int`. What are all the possible outcomes of executing these instructions?

   Core 1: `y = 5/z + w;`

   Core 2: `x = x + y/(w + 1);`

   Core 3: `z = w*(x - y) + z;`

9. (5 points) Sometimes software optimization can dramatically improve the performance of a computer system. Assume that a CPU can perform a multiplication operation in 10 ns, and a subtraction operation in 1 ns. (a) How long will it take for the CPU to calculate the result of d = a × b – a × c? (b) Could you optimize the equation so that it will take less time? (c) If so, what is the resulting speedup?

10. (10 points) In this exercise we look at memory locality properties of matrix computation. The following code is written in MATLAB, where elements within the same column are stored contiguously. Assume each word is a 32-bit integer.

```
for I = 1:8
  for J = 1:8000
    A(I, J) = B(I, 1) + A(J, J);
```

For each variable, indicate whether references to them exhibit spatial or temporal locality. Explain your answers.

11. (10 points) Consider a SEC code that protects 8 bit words with 4 parity bits. If we read the value 0x374, is there an error? If so, correct the error.

12. (10) points Consider an SMT processor that allows instructions from 2 threads to be run concurrently (i.e., there are two functional units), and instructions from either or both threads can be issued to run on any cycle. Assume we have two threads X and Y to run on these CPUs that include the following operations:

Thread X	Thread Y
A1 – takes 3 cycles to execute A2 – no dependences A3 – conflicts for a functional unit with A1 A4 – depends on the result of A3	B1 – take 2 cycles to execute B2 – conflicts for a functional unit with B1 B3 – depends on the result of B2 B4 – no dependences and takes 2 cycles to execute

Assume all instructions take a single cycle to execute unless noted otherwise or they encounter a hazard. How many cycles will it take to execute these two thread? How many issue slots are wasted due to hazards?

13. (10 points) Consider the following MIPS loop:

```
       add   $t2, $zero, $zero
       addi  $s1, $zero, 1
       addi  $t1, $zero, 5
LOOP:  slt   $t4, $zero, $t1
       beq   $t4, $zero, DONE
       addi  $t1, $t1, -1
       addi  $s1, $s1, 2
       mul   $t3, $s1, $s1
       add   $t2, $t2, $t3
       j     LOOP
DONE:
```

What is the value in register $t2 upon completion of the loop execution? Write the equivalent C. Assume that registers `$s1`, `$s2`, `$t1`, and `$t2` are integers `A`, `B`, `i` and `temp`, respectively.

14. (18 points) Consider a pipeline for a register-memory architecture. The architecture has two instruction formats: a register-register format and a register-memory format. There is a single memory addressing mode (offset + base register). There is a set of ALU operations as follows:

ALUop← Rdest, Rsrc1, Rsrc2, offset (Rdest ← Rsrc1 ALUop Rsrc2 or
Rdest ← Rsrc1 ALUop MEM[Rsrc2 + offset])
Rdest ← MEM[Rsrc2 + offset]
MEM[Rsrc2 + offset] ← Rsrc1
where the ALUop is one of the following and Rdest, Rsrc1 and Rsrc2 are registers:
Add, Subtract, And, Or(with or without offset)
Load (Rsrc1 omitted)
Store (Rdest omitted)

Branches use a full compare of two registers and are PC-relative. Assume that this machine is pipelined so that a new instruction is started every clock cycle. The pipeline structure is

IF	RF	ALU1	MEM	ALU2	WB					
	IF	RF	ALU1	MEM	ALU2	WB				
		IF	RF	ALU1	MEM	ALU2	WB			
			IF	RF	ALU1	MEM	ALU2	WB		
				IF	RF	ALU1	MEM	ALU2	WB	
					IF	RF	ALU1	MEM	ALU2	WB

The first ALU stage is used for effective address calculation for memory references and branches. The second ALU stage is used for operations and branch comparisons. RF is both a decode and register-fetch stage. Assume that when a register read and a register write of the same register occur in the same clock cycle, the write data is forwarded. (a) (10 points) For the following code fragment:

```
add     $1, $1, 0($15)
add     $3, $1, 4($15)
add     $2 ,$2, 0($12)
add     $4, $2, 4($1)
store   $1, 8($2)
```

identify the data dependencies and draw a figure (using the multiple clock style) showing them as lines. Assume that there are separate ALUs for the ALU1 and ALU2 pipe stages.

(b) (8 points) Which of these dependencies could be solved by forwarding? Identify the dependencies and describe the forwarding.