The University of Alabama in Huntsville
Electrical & Computer Engineering Department
CPE 431 01
Test 1
September 27, 2011

Name: _____

1. (1 point) A number in floating-point notation that has no leading 0s is said to be _____.
2. (1 point) The address specified in a branch, which becomes the new program counter if the branch is taken is known as the _____ address.
3. (1 point) The _____ instigates a procedure and provides the necessary parameter values.
4. (1 point) An _____.is a computer inside another device used for running one predetermined application or collection of software.
5. (1 point) _____ states that the performance enhancement possible with a given improvement is limited by the amount that the improved feature is used.
6. (9 points) For the MIPS assembly above, assume that the registers `$s0`, `$s1`, `$s2`, `$s3`, contain the values 10, 20, 30, and 40, respectively. Also, assume that register `$s6` contains the value 256, and that memory contains the following values:

Address	Value
256	100
260	200
264	300

Find the value of $s0 at the end of the assembly code.

7. (6 points) What are the binary representations of the opcode, rs, rt, rd, shamt, and funct fields in this instruction?

```
addi $a1, $v0, -6
```

8. (5 points) Show the IEEE 754 binary representation for the floating-point number 307.75_{10} in single precision.

9. (15 points) Consider three different P1, P2, and P3 executing the same instruction set with the clock rates and CPIs given in the following table.

Processor	Clock Rate	CPI
P1	2 GHz	1.3
P2	1.4 GHz	1.0
P3	2.6 GHz	2.2

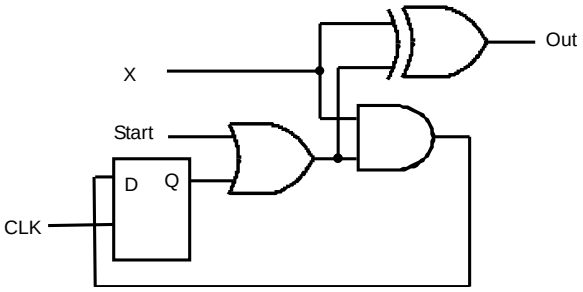
Which processor has the highest performance?

10. (5 points) Suppose the program counter (PC) is set to 0x0000 0020. Is it possible to use the jump (`j`) MIPS assembly instruction to set the PC to the address 0x0000 0021?

11. (10 points) What decimal number does the bit pattern represent if it is a two's-complement integer? An unsigned integer?

    1011 1100 0010 0101

12. (10 points) Consider the following circuit:



Assume that the latency and cost of basic logic elements are as follows:

NOT		AND		OR		XOR		D-element	
Latency	Cost	Latency	Cost	Latency	Cost	Latency	Cost	Latency	Cost
40 ps	1	50 ps	2	60 ps	2	80 ps	3	80 ps	12

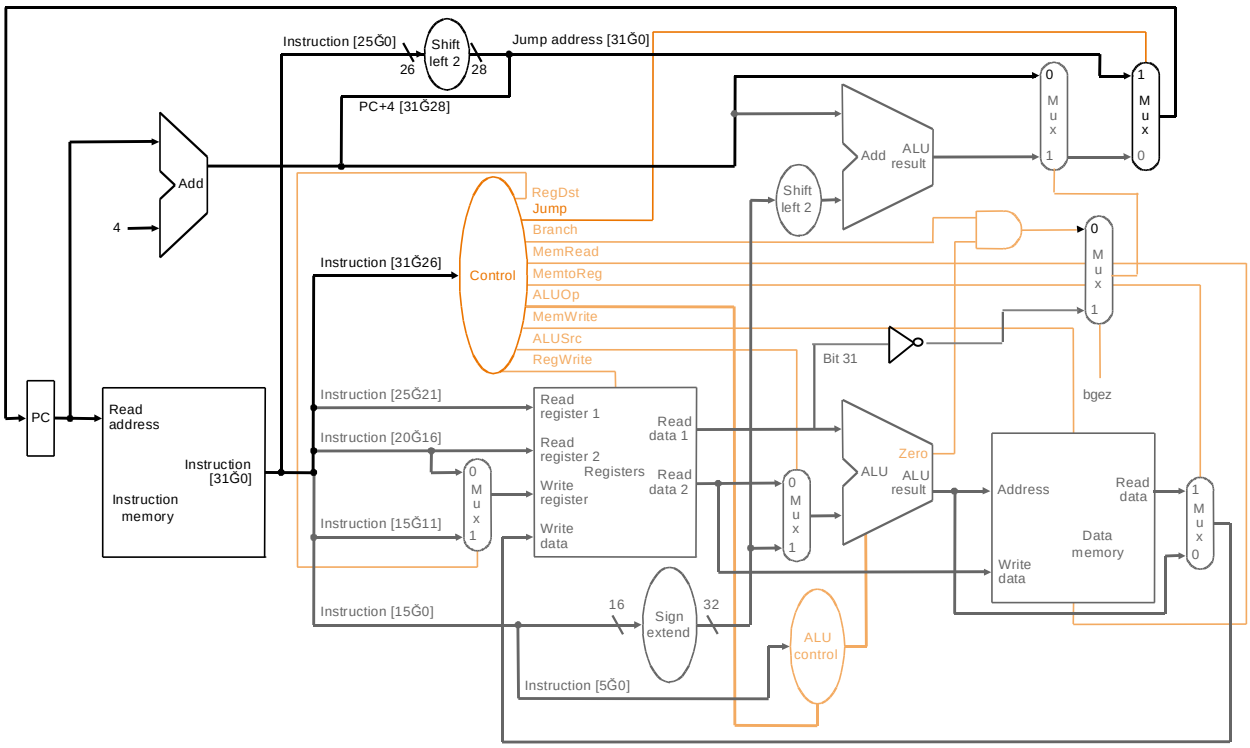
The time given for a D-element is its setup time. The data input of a flip-flop must have the correct value one setup-time before the clock edge (end of clock cycle) that stores that value into the flip-flop. What is the cycle time for the circuit given?

13. (15 points) Add the instruction `bgez` (branch on greater than or equal to zero) to the single-cycle datapath shown in the figure below. The `begz` instruction is defined below. Add any necessary datapaths and control signals and show the necessary additions to the table of control signals given.

`begz rs, label`

if (rs >= 0)
    PC ← PC + 4 + 4*offset
else
    PC ← PC + 4

1	rs	1	offset



Instruction	RegDst	ALUSrc	Memto Reg	Reg Write	Mem Read	Mem Write	Branch	ALUOp1	ALUOp0	bgez
R-format	1	0	0	1	0	0	0	1	0	0
lw	0	1	1	1	1	0	0	0	0	0
sw	d	1	d	0	0	1	0	0	0	0
beq	d	0	d	0	0	0	1	0	1	0
bgez	d	d	d	0	0	0	d	d	d	1

d-don't care

14. (10 points) Assume the following CPIs and the instruction breakdown for executing a given program:

	Instructions (in millions)	CPI
Arithmetic	500	4
Load/Store	300	10
Branch	100	2

Suppose that new, more powerful arithmetic instructions are added to the instruction set. On average, through the use of these more powerful arithmetic instructions, we can reduce the number of arithmetic instructions need to execute a program by 20 %, and the cost of increasing the clock cycle time by only 15 %. Is this a good design choice? Why?

15. (10 points) Your company could speed up a Java program on their new computer by adding hardware support for garbage collection. Garbage collection currently comprises 20% of the cycles of the program. You have two possible changes to the machine. The first one would be to automatically handle garbage collection in hardware. This causes an increase in cycle time by a factor of 1.2. The second would be to provide for new hardware instructions to be added to the ISA that could be used during garbage collection. This would halve the number of instructions needed for garbage collections but increase the cycle time by 1.1. Which of these two options, if either, should you choose?