Question 11 (2 points)

You are given code to handle arithmetic overflow for unsigned numbers, which is labeled handleOverflow. In the following snippet of MIPS code fill in the correct instruction to replace '***', in order to jump to the label correctly .

```
unsignedAdd: addu $v0,$a0,$a1
             nor $t0,$a0,$0
             sltu $t0,$t0,$a1
             *** $t0,$0,handleOverflow
             jr $ra
```

- ◯ bne or bgt
- ◯ bne but not bgt
- ◯ blt
- ◯ beq

Question 12 (2 points)

Consider the following snippet of MIPS code.

```
lui $t0, 0x7fff
ori $t0,$t0,0xfffe
addi $t0,$t0,2
```

Of the statements below, which one is TRUE?

- ◯ Running this snippet will lead to arithmetic overflow, but no arithmetic overflow occurs if the numbers are interpreted as unsigned 32-bit integers.
- ◯ Once the code has been run $t0 contains `0xfffe8001`.
- ◯ Once the code has been run $t0 contains `0x80000000`.
- ◯ Running this snippet will lead to arithmetic overflow, but no arithmetic overflow occurs if the first two rows are interchanged.

Question 13 (2 points)

Consider a comparison between two positive single precision 32-bit IEEE floating point numbers. Which of the following is TRUE?

- ◯ We cannot use an ordinary unsigned integer comparison to test for equality between them, but we can use it to test for 'greater than' or 'less than'.
- ◯ We can use an ordinary unsigned integer comparison to test for equality between them, but not to test for 'greater than' or 'less than'.
- ◯ We can use an ordinary unsigned integer comparison to test for equality between them, as well as to test for 'greater than' or 'less than'.
- ◯ We cannot use an ordinary unsigned integer comparison to test for equality between them, 'greater than' or 'less than'.

Question 14 (2 points)

Consider a simplified IEEE floating point representation where we have 7 bits in total, with 1 sign bit and 3 exponent bits. Which of the following is TRUE?

- ◯ The magnitude of the largest number that can be represented is 224/32.
- ◯ The magnitude of the smallest number that can be represented is 1/32.
- ◯ The magnitude of the difference between the largest and next largest representable numbers is 21/32.
- ◯ The magnitude of the difference between the smallest and next smallest representable numbers is 3/32.

Question 15 (2 points)

In an object file, what does the **relocation** table contain?

- A list of the file's labels and data that can be referenced.
- The binary representation of the data in the source file.
- A list of lines of code that need to be modified later.
- The file's machine code.

Question 16 (2 points)

As covered in this course, what are the classic components found in any computer?

- keyboard, monitor, mouse, processor, and disk
- processor, memory, cache, interrupts, and devices
- compiler, assembler, linker, and loader
- control, datapath, memory, input, and output

Question 17 (2 points)

In an object file, what does the **symbol table** contain?

- The binary representation of the data in the source file.

- The file's machine code.
- A list of lines of code that need to be modified later.

- A list of the file's labels and data that can be referenced.

Question 18 (2 points)

In an object file, what does the **text** segment contain?

- A list of lines of code that need to be fixed later.
- The binary representation of the data in the source file.
- A list of the file's labels and data that can be referenced.
- The file's machine code.

Question 19 (2 points)

When following register conventions in MIPS

- The contents of the registers $a0, $a1, and $a2 must be restored to their original values prior to exiting a subroutine.
- The registers $v0 and $v1 can only be used to pass arguments when calling a subroutine.
- The contents of the registers $s0 through $s7 must be restored to their original values prior to exiting a subroutine.
- It is not possible to have a subroutine which calls itself, since there is only one set of registers which can be used to pass arguments.

Question 20 (2 points)

The design a circuit for a finite state machine proceeds with the following sequence of steps:

- ◯ flip-flop choice → output/next-state equations → state trans. diagram → state trans. table
- ◯ output/next-state equations → flip-flop choice → state trans. diagram → state trans. table
- ◯ state trans. diagram → state trans. table → flip-flop choice → output/next-state equations
- ◯ state trans. table → flip-flop choice → output/next-state equations → state trans. diagram

Question 21 (2 points)

In the vending machine example we discussed in class, there were two inputs D and N, representing the insertion of a dime (10 cents) or a nickel (5 cents), respectively, and a single output G which was to be set to 1 only when at least 15 cents had been inserted. It was possible to reduce the implementation from 9 states to 4 states, by

- ◯ Using a Moore machine.
- ◯ Ignoring the first input to the machine, if it happened to be N = 1.
- ◯ Using a Mealy machine.
- ◯ Letting a state represent the total amount of money that has been received up until that time.

（You can skip this）

Question 22 (Mandatory) (0.0001 points)

Remember that this is a timed exam which is expected to be 3hrs in length. You have 6.5 consecutive hrs (390 consecutive minutes) from the time you start, to complete it. It is open text book, open mycourses material, lecture notes and slides. However, you are not to use logisim or MARS and you are not to consult any external sources or discuss the contents of the exam during the 72 hr period form December 11th 14:00 hrs to December 14th 14:00 hrs.

Each question is multiple choice and there is only one correct answer per question. There are 50 questions in total, each of which is worth 2 points. You can change your response to any question at a later time, should you wish to.

COMP 273 FALL 2020 FINAL EXAM COVER PAGES

Click on the link above to review the two cover pages of the final examination. Then type your name in the space provided for a "written response". By doing so you are affirming that you are signing the statement of academic integrity and that you will comply with all the terms of the final examination.

Question 23 (2 points)

The next question pertains to the following MIPS subroutine steps. This routine calculates the number of different ways to climb a flight of stairs, when at each step one is allowed to either take a single step, or two steps together.

```
steps: addi $sp,$sp,-12
       sw $ra,0($sp)
       sw $s0,4($sp)
       sw $s1,8($sp)

       move $s0,$a0
       bgt $s0,1,count
       move $v0,$s0
       j steps.return

count: addi $a0,$s0,-1
       jal steps
       move $s1,$v0

       addi $a0,$s0,-2
       jal steps
       add $v0,$v0,$s1

steps.return:
       lw $ra,0($sp)
       lw $s0,4($sp)
       lw $s1,8($sp)
       addi $sp,$sp,12
       jr $ra
```

After executing the following two lines of code

```
li $a0, 5
jal steps
```

how many times will steps be called?

- ◯ 9
- ◯ 25
- ◯ 15
- ◯ 1

Question 24 (2 points)

After running the following snippet of MIPS code, which of the subsequent statements is TRUE?

```
sra $t0,$s1,31
xor $s0,$s1,$t0
subu $s0,$s0,$t0
```

- ◯ $s0 contains (1 + $s1).
- ◯ $s0 contains the absolute value of $s1.
- ◯ $s0 contains (1 - $s1).
- ◯ $s0 contains the difference between $s1 and $s0.

Question 25 (2 points)

The next question pertains to the following MIPS subroutine Goo, which assumes that the value of a non-negative integer n has been passed via the argument register $a0.

```
Goo: addi $sp,$sp,-8
     sw $ra,4($sp)
     sw $a0,0($sp)

     slti $t0,$a0,1
     beq $t0,$zero,L1
     addi $v0,$zero,1
     addi $sp,$sp,8
     jr $ra

L1:  addi $a0,$a0,-1
     jal Goo

     lw $ra,4($sp)
     lw $a0,0($sp)
     mul $v0,$v0,$a0
     addi $sp,$sp,8
     jr $ra
```

Which of the following statements is false?

- The subroutine Goo will terminate.
- There is no need for the subroutine Goo to save $ra.
- The subroutine Goo uses the stack.
- The subroutine Goo is recursive.

Question 26 (2 points)

Which of the following is a TRUE MIPS instruction?

- `sw $ra, 0($sp)`
- `move $t0, $t1`
- `li $s0, 5`
- `addi $t0, $t1, 40000`

Question 27 (2 points)

Consider the MIPS procedure Moo below. Which snippet of pseudo-code, among the choices below (where the input is an integer n), best represents what it does?

```
Moo:    bltz  $a0,Label
        move  $v0,$a0
        jr $ra

Label:  addiu $v0,$a0,1
        jr $ra
```

- if (n < 0) return n++; else return n++;
- if (n < 0) return ++n; else return n++;
- if (n < 0) return n++; else return ++n;
- if (n < 0) return ++n; else return ++n;

Question 28 (2 points)

A finite state machine has one input, X, and two outputs, O1 and O2. The output O1 should be set to 1 every time the input sequence 101 is observed, provided that the input sequence 011 has never been seen. Otherwise O1 should be set to 0. The output O2 should be set to 1 every time the input sequence 011 is observed. Otherwise it should be set to 0. Finally, once O2 has been set to 1, O1 can never be set to 1. What is the minimum number of states required to implement this machine using a Mealy design?

- 8
- 9
- 10
- 7

Question 29 (2 points)

Consider a sequential circuit that has a single input X and a single output Z. The behavior of the circuit is as follows. The output Z should be a 1 when the input sequence 1010 has been detected, and should be 0 at all other times. It can be assumed that patterns may overlap, e.g., the sequence 101010 should result in the output Z being asserted twice. If you are free to choose either a Mealy or a Moore machine, the minimum number of states in which this machine could be constructed is

- 7
- 6
- 5
- 4

Question 30 (2 points)

Consider a program which carries out a fixed number of iterations where it takes very large steps through a 1D array, where for each iteration the array doubles in size. Which of the following statements is correct?

- The runtime of this program is approximately flat initially, following which it increases, following which it later increases again dramatically.
- The runtime of this program doubles every iteration, because the array has doubled in size.
- The runtime of this program increases linearly as a function of the iteration number.
- None of the above.

Question 31 (2 points)

Consider a large array 1D integer A with 128 × 1024 × 1024 elements. Compared to program 1, which loops through the array visiting every element in succession, program 2 which loops through the array visiting every 25th element in succession, has the following property.

- ◯ It will be less than 25 times as fast because spatial locality is violated.
- ◯ It is approximately 25 times faster, since it has 25 times less work to do.
- ◯ It will be be less than 25 times as fast because it will always result in queries from L1 cache but never from L2 or L3 cache.
- ◯ It is slightly more than 25 times faster, since it has 25 times less work to do and it results in more cache hits.

Question 32 (2 points)

What is the **average memory access time** in a two level cache, assuming the L1 hit time is 1 cycle, the L1 miss rate is 5%, the L2 hit time is 5 cycles, the L2 miss rate is 35%, and the L2 miss penalty is 100 cycles?

- ◯ 1 cycle
- ◯ 6 cycles
- ◯ 3 cycles
- ◯ 2 cycles

Question 33 (2 points)

Which of the following statements about unmapped regions of MIPS virtual memory is TRUE?

- ◯ Unmapped virtual addresses are in the kernel segment of the virtual memory.
- ◯ The physical address of unmapped regions is the same as their virtual address.
- ◯ The three most significant bits of an unmapped virtual address differ from the corresponding physical addresses.
- ◯ All of the above.

Question 34 (2 points)

Given a 32 KB cache (i.e., 2^{15} bytes) with 8 word blocks, and assuming memory addresses are specified by 32-bits, how big are the **tag**, **index**, and **offset** for a direct-mapped cache?

- ◯ Tag has 12 bits, index has 15 bits, offset has 5 bits.
- ◯ Tag has 12 bits, index has 5 bits, offset has 15 bits.
- ◯ Tag has 12 bits, index has 5 bits, offset has 10 bits.
- ◯ Tag has 17 bits, index has 10 bits, offset has 5 bits.

Question 35 (2 points)

The correct conversion of the hexadecimal number `0x83B06000` to octal (base 8) is

- ◯ `40730140000`
- ◯ `20354060000`
- ◯ `10354060000`
- ◯ `40730140001`

Question 36 (2 points)

Which of the following is a correct conversion of -11 from decimal to 8 bit 2's complement binary?

- 11110101
- 11110011
- 11110100
- 11110010

Question 37 (2 points)

Suppose you have an n-bit integer in base 6. Which of the following statements is TRUE?

- If the integer is signed (that is 6's complement) the largest value the integer can have is 6^{n-1}.
- If the integer is signed (that is 6's complement) the smallest value the integer can have is $-5 \times 6^{n-1}$.
- If the integer is unsigned, the largest value the integer can have is 6^n.
- If the integer is unsigned, the smallest value the integer can have is 1.

Question 38 (2 points)

To represent the decimal number 7 in single precision IEEE 754 floating point, what would you put in the 8 bit exponent field?

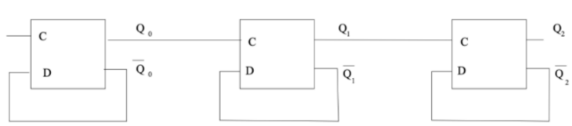
- 11111110
- 10000000
- 00000010
- 10000001

Question 39 (2 points)

Consider the circuit shown below, which contains 3 D flip-flops. The C input of the flip-flop on the left is connected to a clock. Assume that each D flip-flop is a falling edge triggered flip-flop. Assume that initially the outputs

$$Q_2Q_1Q_0$$

are 000.



Then, after the next successive falling edges of the clock the outputs

$$Q_2Q_1Q_0$$

will change to

- 001, 010, 011, 100, 101, 110, 111, ...
- 111, 110, 101, 100, 011, 010, 001, ...
- 101, 010, 101, 010, 101, 010, 101, ...
- 111, 000, 111, 000, 111, 000, 111, ...

Question 40 (2 points)

Of the choices below, what best describes a flip flop?

- It is a set of registers.
- It is a clock that cycles between HIGH and LOW.
- It is a cell capable of storing one bit of information.
- It is an analog device capable of providing synchronization.

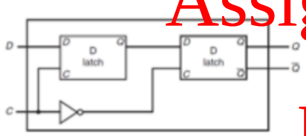
Question 41 (2 points)

Which of the following describes the operation of a rising edge triggered D flip-flop?

- When both the D and clock inputs are LOW, an invalid state exists.
- The output will follow the input on the leading edge of the clock.
- If the D and clock inputs are both HIGH, the output will toggle.
- The input is toggled into the flip-flop on the leading edge of the clock and is passed to the output on the trailing edge of the clock.

Question 42 (2 points)

A falling-edge D flip flop is constructed from D latches as shown in the diagram below. What best describes the construction of a **toggle flip flop** from a D flip flop, and its correct operation?



- Connect Q to D and toggle Q by having C go from LOW to HIGH to LOW.
- Connect $\bar{Q}$ to D and toggle Q by having C go from LOW to HIGH to LOW.
- Connect Q to C and toggle Q by having D go from LOW to HIGH to LOW.
- Connect $\bar{Q}$ to C and toggle Q by having D go from LOW to HIGH to LOW.

Question 43 (2 points)

An R-S latch has the following property:

- It has a forbidden state, which is impossible to reach.
- It can be built using only two OR gates and appropriate connections.
- The value of its output Q is always determined only by its present R and S inputs.
- With additional circuity it can be turned into a D latch.

Question 44 (2 points)

A sequential circuit built with D flip-flops

- Cannot be converted into an equivalent circuit which uses both D flip-flops and J-K flip-flops.
- Is always simpler in terms of total number of gates than an equivalent circuit built with J-K flip flops.
- Cannot always be converted into an equivalent circuit built with T flip-flops.
- Can be converted into an equivalent sequential circuit built with J-K flip flops.

Question 45 (2 points)

What is the Boolean algebra rule that states

$$\overline{(A+B)} = \overline{A}.\overline{B}?$$

- ◯ commutativity
- ◯ associativity
- ◯ De Morgan's Law
- ◯ distributivity

Question 46 (2 points)

Which register will remain unchanged after a call to a subroutine according to register conventions?

- ◯ $v0
- ◯ $t0
- ◯ $a0
- ◯ $s0

Question 47 (2 points)

With s0 corresponding to register $s0, and s1 corresponding to register $s1, what implements the code if (s0 >= s1) goto label?

◯
```
slt $t0, $s0, $s1
bne $t0, $0, label
```

◯
```
slt $t0, $s0, $s1
beq $t0, $0, label
```

◯
```
slt $t0, $s1, $s0
bne $t0, $0, label
```

◯
```
slt $t0, $s1, $s0
beq $t0, $0, label
```

Question 48 (2 points)

Which of the following MIPS instructions will modify the contents of the $ra register?

- ◯ `bne $ra, $t1, label`
- ◯ `jr $ra`
- ◯ `sw $ra, 0($sp)`
- ◯ `jal label`

Question 49 (2 points)

If $t0 equals `0xb01dface`, what will be in $t1 after the instruction

`sra $t1, $t0, 4`?

- ◯ `0xeb01dfac`
- ◯ `0x0b01dfac`
- ◯ `0xfb01dfac`
- ◯ `0x01dface0`

Question 50 (2 points)

Which of the following instructions computes the negative of the integer stored in $t0?

- ◯ `xor $t0, $0, $t0`
- ◯ `sub $t0, $0, $t0`
- ◯ `nor $t0, $t0, $t0`
- ◯ `nor $t0, $t0, $0`

Question 51 (2 points)

Assume that states in a finite state machine are represented using JK flip-flops. The new state Q+ expressed in terms of the present state Q and the inputs J and K, is

- ◯ $Q^+ = J.Q + K.\overline{Q}$

- ◯ $Q^+ = J.Q + K$
- ◯ $Q^+ = J + \overline{K}.\overline{Q}$
- ◯ $Q^+ = J.\overline{Q} + \overline{K}.Q$