Review Test Submission: Exam I - Fall 2019

User	Chaoran Li
Course	(MERGED) CS 3340.501 - SE 3340.501 - F19
Test	Exam I - Fall 2019
Started	9/26/19 5:51 PM
Submitted	9/26/19 7:11 PM
Status	Completed
Attempt Score	68.25 out of 100 points
Time Elapsed	1 hour, 20 minutes out of 1 hour and 20 minutes
Instructions	• Exam time is 75 minutes, and a 5 mins extension is allowed if needed. A time will expire after 80 mins and your work will be submitted automatically. • Only **handwriting notes and the MIPS reference card** are allowed to use. No other printed materials, no books, **no slides**! • The use of the software calculator available on the top of the Respondus Lockdown Browser window is allowed. • An answer to an essay question must be concise, up-to-the-point and must cover all necessary ingredients to earn full credit. • Multiple answer questions have at least one correct and at least one incorrect answer. **Select all correct answers and avoid incorrect answers, as incorrect answers have a penalty. Do NOT select all answers, no credit will be given if all answers are selected.** • **Incorrect answers to a matching question also have a penalty.** • **Read the questions carefully!!!** • **Good luck!**
Results Displayed	All Answers, Submitted Answers, Correct Answers, Feedback, Incorrectly Answered Questions

Question 1

2 out of 2 points

A RISC architecture processor has no instruction that directly operates on data stored in the main memory.

Selected Answer: True

Answers: True

False

Question 2

5 out of 5 points

If a programmer wants to load the 32-bit word 0xBABEFACE into register $t0, the following sequence of MIPS instructions can be used:

```
                    [a] $t0, [b]            # assign
the upper part

                    [c] $t0, [d], [e]       # assign
the lower part
```

Note: do not forget to include '0x' for hexadecimal numbers in your answers.

Specified Answer for: a ✓ lui

Specified Answer for: b ✓ 0xBABE

Specified Answer for: c ✓ ori

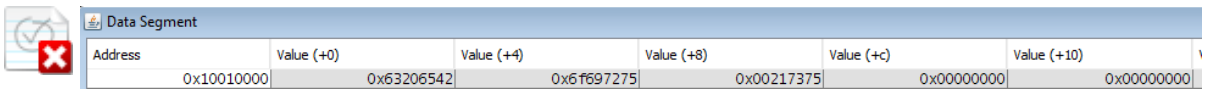
Specified Answer for: d ✓ $t0

Specified Answer for: e ✓ 0xFACE

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	lui	Case Sensitive
Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	0xbabe	
Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	ori	Case Sensitive
Correct Answers for: d		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	$t0	Case Sensitive
Correct Answers for: e		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	0xface	

Question 3

0 out of 5 points



A string is stored in the data segment starting at address 0x10010000 of the memory of a MIPS-32 processor-based computer system using the NULL-terminated method, and a portion of the data segment is shown above. Also, this system uses the little-Endian scheme.

The stored string is "_____".

Note: make sure that your answer contains no extra letters and the characters must be exact as stored in the memory, including cases.

Selected Answer: ✗ c eBoiru

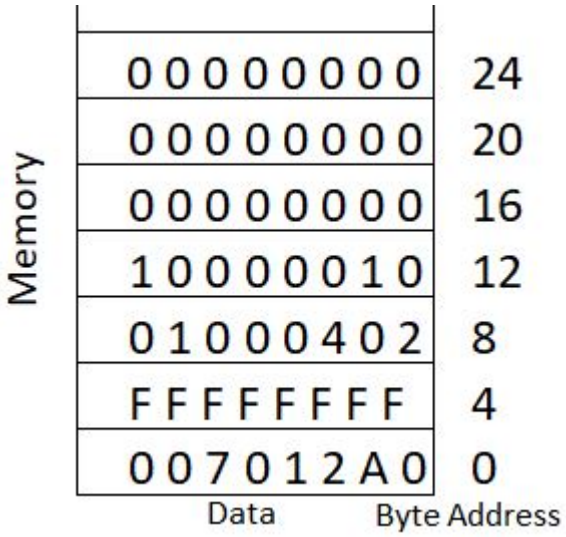
Correct Answer:

Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	Be curious!	*Case Sensitive*

Question 4

0 out of 6 points

A portion of the memory of a MIPS-32 based computer is shown below, and the computer is using the little Endian scheme to store data in memory.



After the processor has execute the following instructions:

```
add  $s0, $zero, $zero
lb   $t0, 9($s0)
sb   $t0, 5($s0)
```

the content of register $t0 is 0x**[a]**, and the memory word starting from byte address **[b]** is changed to 0x**[c]**.

If the computer was using the big-Endian scheme to store data memory, the content of register $t0 is 0x**[d]**, and the memory word starting from byte address **[e]** is changed to 0x**[f]**.

Note: do not omit leading 0s, and do not include '0x' in your answers.

Specified Answer for: a ✗ 10010004

Specified Answer for: b ✗ 5

Specified Answer for: c ✗ 10010004

Specified Answer for: d ✗ 10000000

Specified Answer for: e ✗ 5

Specified Answer for: f ✗ 10000000

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	00000004	
Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	4	
Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	FFFF04FF	
Correct Answers for: d		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	00000000	

Correct Answers for: e		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	4	

Correct Answers for: f		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	FF00FFFF	

Question 5

0 out of 4 points

In the main memory of a computer system that uses a MIPS processor

Selected Answers:
- a. both data and instructions are stored as hexadecimal digits.
- b. a data word (e.g. an integer) can locate between two instruction words.
- c. a data word (32-bits) does not have any inherent meaning and may contain an integer, a single-precision floating-point number, or a portion of a string.

Answers:
- a. both data and instructions are stored as hexadecimal digits.
- b. a data word (e.g. an integer) can locate between two instruction words.
- c. a data word (32-bits) does not have any inherent meaning and may contain an integer, a single-precision floating-point number, or a portion of a string.
- d. a 32-bit word must be stored at a memory location whose address is a multiple of 4.

Question 6

2 out of 2 points

In the Von Neumann computer architecture, the content of the Program Counter register (PC) determines the control flow of a program.

Selected Answer: True

Answers: True
False

Question 7

10 out of 10 points

SuperNanoPi is a nano-processor that operates on 3-bit numbers. Fill out the following number representation table for SuperNanoPi in ***decimal values*** for various number representations schemes.

Binary	Unsigned	Sign-magnitude	2's complement
000	0	0	0
001	**[a]**	1	1
010	**[b]**	**[e]**	2
011	**[c]**	3	3
100	4	-0	**[h]**
101	5	-1	-3

110	6	**[f]**	**[i]**
111	**[d]**	**[g]**	**[j]**

Specified Answer for: a 1

Specified Answer for: b 2

Specified Answer for: e 2

Specified Answer for: c 3

Specified Answer for: h -4

Specified Answer for: f -2

Specified Answer for: i -2

Specified Answer for: d 7

Specified Answer for: g -3

Specified Answer for: j -1

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	1	
Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	2	
Correct Answers for: e		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	2	
Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	3	
Correct Answers for: h		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	-4	
Correct Answers for: f		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	-2	
Correct Answers for: i		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	-2	
Correct Answers for: d		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Pattern Match	7	
Correct Answers for: g		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	-3	
Correct Answers for: j		
Evaluation Method	**Correct Answer**	**Case Sensitivity**

Exact Match	-1	

Question 8

0 out of 6 points

Consider the following MIPS assembly language program:

```
#  Data area
           .data
mydata:    .word 0x1CAFE, 0x4DAD
#
output:    .word 0              # for storing the result
#
#  Instructions (program) area
           .text
main:      la  $s0, mydata      # load the address of memory
location mydata to $s0
           lw  $t0, ($s0)       #
           lw  $t1, 4($s0)      #
           sub $t2, $t0, $t1    #
#
           la  $s1, output      # Load the address of memory
location output to $s1
           sw  $t2, ($s1)       #
#
           li  $v0, 1           # syscall #1 (print int)
           srl $a0, $t2, 1      # divide by 2
           syscall              # invoke syscall to print

#

# The END
```

1) What will be seen on MARS console when this program runs? [x]

Selected Answer: 7146

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	48808	

Question 9

0 out of 4 points

Convert the following decimal numbers to hexadecimal numbers to fill the blanks, assuming they are stored in main memory of a MIPS-32 processor-based computer. Do NOT omit the leading 0s, and do NOT include '0x' in your answers.

a. 1776 = 0x**[a]**

b. -17 = 0x**[b]**

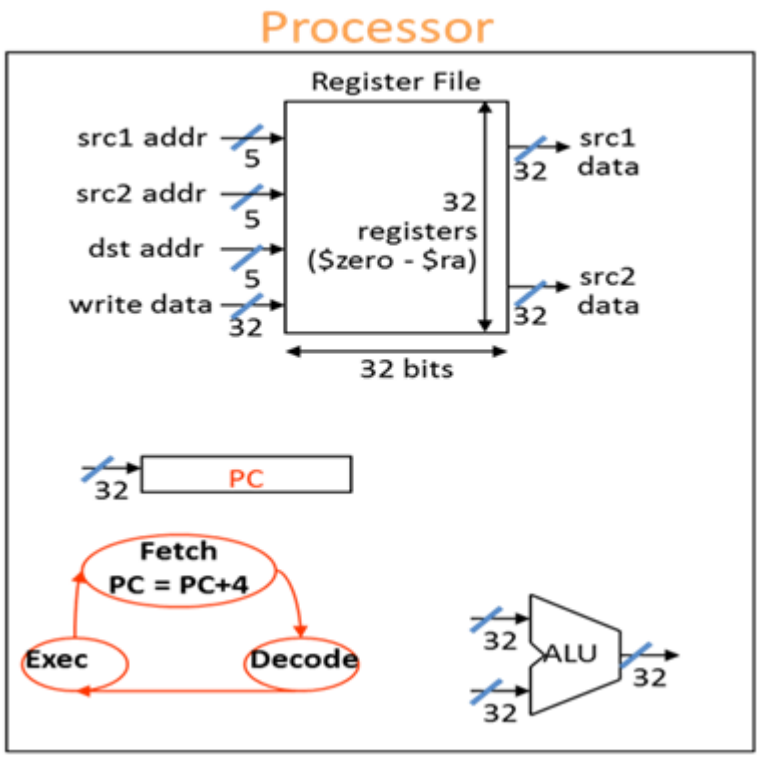
Specified Answer for: a ✗ 06f0

Specified Answer for: b ✗ 1101

Correct Answers for: a		
Evaluation Method	Correct Answer	Case Sensitivity
✓ *Exact Match*	000006f0	
Correct Answers for: b		
Evaluation Method	Correct Answer	Case Sensitivity
✓ *Exact Match*	FFFFFFEF	

Question 10

3 out of 5 points (Extra Credit)



The diagram above shows the major functional elements of a processor.

Describe HOW does the processor use the element named 'Register File' in the diagram. The inputs (e.g. src1 addr) and the outputs (e.g. src1 data) must be included in the description to earn full credits.

Selected Answer:	I would use a R format as an example. 1. Register File will get a 32-bit src1 data according to 5-bit src1 address of a register. 2. Register File will get a 32-bit src2 data according to 5-bit src2 address of a register. 3. Then use ALU to calculate with src1 data and src2 data and generate the 32-bit write data. 4. Register File will save 32-bit write data according to 5-bit dst address which represents a register.
Correct Answer:	[None]
Response Feedback:	[None Given]

Question 11

3 out of 4 points

A program module written in a high-level language, such as C++, must be translated into a **[a]** code module that contains the binary representation of data and instructions by a system program called **[b]**. Another system program named **[c]** then combines several resulting modules, and code from system libraries to create an executable file. This executable file then can be put into the main memory of a computer system by a system program called **[e]** so that the processor can execute instructions in that executable file.

Specified Answer for: a assemble

Specified Answer for: b compiler

Specified Answer for: c linker

Specified Answer for: e loader

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	machine	
Exact Match	object	
Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Contains	compiler	
Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Contains	linker	
Correct Answers for: e		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Contains	loader	

Question 12

4 out of 4 points

There are many system programs that are used in a computer system.

Selected Answers: A linker combines many object files into one executable file.
a.
b.
A compiler may translate a high-level language program directly to machine code without producing assembly language files.

Answers: A linker combines many object files into one executable file.
a.
b.
A compiler may translate a high-level language program directly to machine code without producing assembly language files.

c.
An assembler can produce an executable program that can run on a computer.

Examples of system programs: assembler, linker, compiler, editor.
d.

Question 13

2 out of 2 points

Given that

$$\text{CPU Time} = \text{CPU Clock Cycles} \times \text{Clock Cycle Time} = \frac{\text{CPU Clock Cycles}}{\text{Clock Rate}}$$

What techniques can be used to improve the performance of a processor?

Selected Answer:	performance = 1 / execution time To improve the performance, we should reduce the CPU Time. That is reduce the CPU Clock Cycles, or reduce the Clock Cycle Time(increase the Clock Rate).
Correct Answer:	[None]
Response Feedback:	[None Given]

Question 14

4 out of 4 points

In the data segment of a memory of a MIPS-32 processor-based computer, a memory word is used to store a single-precision floating-point number.

Its value in decimal is on the left, write their corresponding bit pattern as stored in memory in hexadecimal in the blank on the right.

Note: do not include '0x', and do not omit leading or trailing 0s!

0.625 **0x[b]**

Specified Answer for: b 3f200000

Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	3F200000	

Question 15

4 out of 4 points

A word in the text segment of the memory of a computer system using a MIPS-32 processor has the following value.

0x01204027

This word represents the following MIPS instruction:

[a] **[b], [c], [d]**

Specified Answer for: a nor

Specified Answer for: b $t0

Specified Answer for: c $t1

Specified Answer for: d ✓ $zero

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	nor	Case Sensitive
Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	$t0	Case Sensitive
Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	$t1	Case Sensitive
Correct Answers for: d		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
✓ *Exact Match*	$zero	Case Sensitive
✓ *Exact Match*	$0	

Question 16

0 out of 3 points

A user runs a program on a MIPS-32 processor-based computer system and the clock frequency of the processor is 2 GHz. After this program is completed the system reports that 20 billion instructions were executed, and the average CPI (Cycle-Per-Instruction) is 0.4. What is the CPU time (in seconds) spent on this run?

Selected Answer: ✗ 0.004
Correct Answer: ✓ 4
Answer range +/- 0 (4 - 4)

Question 17

4.25 out of 5 points

MIPS instructions can be grouped into various categories depending on what the instruction does. Match MIPS instructions with their category. In case instructions match with two or more categories choose the most fitting one.

Question	Correct Match	Selected Match
addi	✓ b. arithmetic and logical	✓ b. arithmetic and logical
ori and sll	✓ b. arithmetic and logical	✓ b. arithmetic and logical
j and jr	✓ f. change control flow unconditionally (jumping)	✓ f. change control flow unconditionally (jumping)
beq and bne	✓ c. change control flow conditionally (branching)	✓ c. change control flow conditionally (branching)
lb and sb	✓ d. data transfer	✓ d. data transfer
slt and beq	✓ c. change control flow conditionally (branching)	✓ c. change control flow conditionally (branching)
ll and sc	✓ e. synchronization	✓ e. synchronization

lbu	✓ d. data transfer	✓ d. data transfer
sub	✓ b. arithmetic and logical	✓ b. arithmetic and logical
lw and sw	✓ d. data transfer	✗ b. arithmetic and logical

All Answer Choices

a. input/output

b. arithmetic and logical

c. change control flow conditionally (branching)

d. data transfer

e. synchronization

f. change control flow unconditionally (jumping)

g. floating point

Question 18

8 out of 12 points

Assuming the following MIPS assembly language code snippet is loaded into the memory of a MIPS-32 processor-based system starting at memory address 0x00400000.

Fill the blanks on the right with the corresponding machine code in hexadecimal.

```
      addi $t1, $zero, 200   # 0x[a]
Loop: sub  $s1, $s2, $s6     # 0x[b]
      sw   $t1, 0($s1)       # 0x[c]
      beq  $t0, $s5, Exit    # 0x[d]
      addi $t1, $t1, -1      # 0x[e]
      j    Loop              # 0x[f]
Exit:
```

Note: ***Do NOT include '0x' and do NOT omit the leading '0's in your answers.***

Specified Answer for: a ✓ 200900C8

Specified Answer for: b ✓ 02568822

Specified Answer for: c ✗ ad310000

Specified Answer for: d ✓ 11150002

Specified Answer for: e ✓ 2129FFFF

Specified Answer for: f ✗ 08010001

Correct Answers for: a

Evaluation Method	Correct Answer	Case Sensitivity
✓ *Exact Match*	200900c8	

Correct Answers for: b		
Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	02568822	
Correct Answers for: c		
Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	ae290000	
Correct Answers for: d		
Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	11150002	
Correct Answers for: e		
Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	2129ffff	
Correct Answers for: f		
Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	08100001	

Question 19

3 out of 4 points

Pseudo instructions in MIPS assembly language are

Selected Answers:
- b. translated into one or more 32-bit word machine code in memory.
- c. understood by a MIPS processor.
- d. used to narrow the semantic gap between programmers and the ISA.

Answers:
- a. used by linkers.
- b. translated into one or more 32-bit word machine code in memory.
- c. understood by a MIPS processor.
- d. used to narrow the semantic gap between programmers and the ISA.

Question 20

4 out of 4 points

The following MIPS branching pseudo instruction

```
   bge $s0, $s1, Label  # branch to Label if $s0 is greater or equal $s1
```

will be translated to these two native instructions by the MIPS assembler:

```
    slt $at, $s0, $s1

    [a] $at, [b], Label
```

Specified Answer for: a beq

Specified Answer for: b $zero

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	beq	

Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	$zero	
Exact Match	$0	

Question 21

6 out of 6 points

High-level language constructs such as loops can be compiled to assembly language code using control-flow changing instructions such as jumping and branching.

For example the following code snippet

```
while (i != 100)
    i = i + 1;
```

can be compiled to the following code, assuming **i** is stored in register **$s0** of a MIPS-32 processor.

```
    addi $t0, $zero, 100 # $t0 contains 100
L0: [a]  $s0, [b], [c]   # if i == 100 skip
    addi  $s0, $s0, [d]  # update i
    [e]  [f]             #
L1: ...
```

Fill the blanks to complete the assembly language code.

Specified Answer for: a beq

Specified Answer for: b $t0

Specified Answer for: c L1

Specified Answer for: d 1

Specified Answer for: e j

Specified Answer for: f L0

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	beq	Case Sensitive

Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	$t0	Case Sensitive

Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**

Exact Match	L1	Case Sensitive
Correct Answers for: d		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	1	
Correct Answers for: e		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	j	Case Sensitive
Correct Answers for: f		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	L0	Case Sensitive

Question 22

4 out of 4 points

A MIPS assembly language program has the following code snippet:

```
DotheMath: add  $t0, $s0, $s0
           add  $t0, $t0, $t0
```

This code snippet can be replaced by one MIPS instruction below. Fill the blanks to complete that instruction.

```
DotheMath: [a]  [b], [c], [d]
```

Specified Answer for: a sll

Specified Answer for: b $t0

Specified Answer for: c $s0

Specified Answer for: d 2

Correct Answers for: a		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	sll	
Correct Answers for: b		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	$t0	Case Sensitive
Correct Answers for: c		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	$s0	Case Sensitive
Correct Answers for: d		
Evaluation Method	**Correct Answer**	**Case Sensitivity**
Exact Match	2	

Response Feedback: sll can be used to multiply by 2^k. k = 2 means multiply by 4.

Wednesday, November 13, 2019 9:48:37 PM CST

← OK