Final Review for Chapters 1 and 2

These images should help in analyzing the problems.
Use upper case when typing an instruction or a HEX value.
Put ONE space between operands in the instructions

Name	Format	Example					
ADD	R	1112	3	0		2	1
SUB	R	1624	3	0		2	1
ADDI	I	580	100			2	1
SUBI	I	836	100			2	1
LDUR	D	1986	100		0	2	1
STUR	D	1984	100		0	2	1
Field size		11 or 10 bits	5 bits	5 or 4 bits	2 bits	5 bits	5 bits
R-format	R	opcode	Rm	shamt		Rn	Rd
I-format	I	opcode	immediate			Rn	Rd
D-format	D	opcode	address		op2	Rn	Rt

Name	Fields						
Field size		6 to 11 bits	5 to 10 bits	5 or 4 bits	2 bits	5 bits	5 bits
R-format	R	opcode	Rm	shamt		Rn	Rd
I-format	I	opcode	immediate			Rn	Rd
D-format	D	opcode	address		op2	Rn	Rt
B-format	B	opcode	address				
CB-format	CB	opcode	address				Rt
IW-format	IW	opcode	immediate				Rd

Mnemonic	Format	Width	Binary
LDUR	D	11	11111000010
STUR	D	11	11111000000
MOVK	IM	9	111100101
BR	R	11	11010110000
LSL	R	11	11010011011
LSR	R	11	11010011010
MOVE	IM	9	110100101
SUBI	I	10	1101000100
SUB	R	11	11001011000
CBNZ	CB	8	10110101
CBZ	CB	8	10110100
BL	B	6	100101
ANDI	I	10	1001001000
ADDI	I	10	1001000100
ADD	R	11	10001011000
AND	R	11	10001010000
B.Cond	CB	8	01010100
B	B	6	000101

1 16 points

Associate the line of code with the category

Triple the value of X1

Multiply by 33

Create the 2s complement of register X3

Integer divide by 8

Possible answers

- ADD X3, X3, X3, LSL 5
- MULI X1, X1, 3
- Multiply X3 by 16
- Calculate the Reminder
- LSR X3, X3, #3
- MULI X3, X3, #-1
- XORI X3, X3, #-1. followed by ADDI X3, X3, #1
- DIVI X3, X3, #8
- ADD X1, X1, X1, LSL 1
- MULI X3, X3, 33

2 15 points

We need to call procedure **calculateBonus**
A - What line of code will call our procedure

You have a procedure that needs to save 3 registers X2, X3, and X6. These registers are 64 bit registers.
When you enter the procedure you need to modify the stack to allocate space and to save these registers. put registers in order from lowest to highest
B - What lines of code would be used to create the space on the stack.

// need space for 3 64 bit registers

// save register X2

// save register X3

// save register X6

C - at the end of the procedure the registers and the stack need to be restored. What are those lines of code?

D - What is the line of code to return from a procedure

// return from procedure

3 8 points

List 4 different kinds of **R - Type** instructions.
Include any registers, addresses or immediate values needed. Make it look like a real line of code.

Edit View Insert Format Tools Table

12pt Paragraph

p

4 14 points

This table lists either the Op Code or the Operation to be executed. Either Determine the Hex value for the instruction or figure out what Operation is to be performed. Use ALL CAPS for the HEX or the OPERATOR and the REGISTER

Op Code	Operator
D37DF02A	, ignore shamt
8B0A000A	ADD x10, x0, x10
F8400149	
	LDUR x11, [x10, #8]
	STUR x11, [x10, #0]
F8008149	
	BR x30

5 20 points

Write the code to support a WHILE loop.
Register X4 must be initialized to 0 by you,
Register X5 will be the loop index and you must initialize it to 19

```
let   max = 19;
let index = 0;
while (index < max) {
    // whatever
    index++;
}
```

Edit View Insert Format Tools Table

12pt Paragraph

p

6 10 points

Given this line of Code (which is an R Format) what are the values for the respective fields

	opcode	immediate	Rn	Rd	Hex Value
SUBI SP, SP, #0x30					

7 12 points

For the following convert HEX to unsigned Decimal and Decimal to HEX. Use capital letters for A-F and no commas for the decimal numbers

1. Convert 0x1234 to decimal
2. Covert 1234 to Hex
3. Convert -555 to 16-bit Binary. (answer will contain 16 bits)
4. Convert 0000 1110 0011 0011 to Decimal.
5. Convert (assume we have an 8-bit integer) 1110 0111 from SIGNED Binary and to UNSIGNED

8 5 points

What are the classic components of a computer

- [ ] silicon
- [ ] Output
- [ ] Memory
- [ ] vacuum tubes
- [ ] electricity
- [ ] Control
- [ ] Input
- [ ] Datapath

9 5 points

Match the abbreviation with the measurement

s	
ms	
µs	
ns	
ps	

Possible answers

bs | 1 Trillionth of a second | fs | 1 second | 1 Billionth of a second | 1 Millionth of a second | 1 Thousandth of a second | ts

10 10 points

The cells in Yellow are steps along the way which you will need to solve for the Green cells.
Do NOT include commas or spaces, If the answer does not have a decimal portion do not include it. DO include the leading zero is answer is less than 1
BAD BAD BAD
12,000,000 12.0 12.10 12 000 000 .4 0.40
GOOD
12000000 12 12.1 12000000 0.4 0.4

Processor B runs App X					

		Run Time	**0.0002 secs**		
# of Cycles	**# of Instructions**	**Instr Type**	Cycles per Instructions	% of Instructions	CPI for Instruction Type
	500,000	**Add/Mov Instr**	**1**		
	250,000	**Branch**	**2**		
	500,000	**Multiply**	**3**	0.4	1.2
Total Cycles	Total # of Instructions				Average CPI
	Clock Rate GHz	Clock Rate (cycles / sec)			
	Cycle Time (Sec/Cyc) x 10 ^-12	Clock Cycle Time (sec / cycle)			

11 8 points

Given three pieces of information about the application parameters calculate the missing piece. If number is less then 1 include the leading 0. Only include the significant fractional portion, no trailing zeros. Do not include commas or spaces.
0.5, 1.55, 2, 1.7

CPI	# of Instructions	Clock Rate Cycles/Sec	CPU Time
2.25	200000000	900000000	0.5
3	100000000	1000000000	
1.5	100000000		0.3
1.5	125000000		0.75
3.5		350 000 000	1.25
4		5 000 000	120
	150 000 000	5 000 000	66
	12 000	240 000	0.3

12 14 points

Order chips from slowest to fastest

Slowest

- 5 ns seconds per cycle
- 5×10^{-15} seconds per cycle
- 200 ns seconds per cycle
- 450×10^{-12} seconds per cycle
- 50.0 Ghz cycles per second
- 1.8 GHz cycles per second
- 10×10^{-15} seconds per cycle

fastest

13 12 points

The cells in Yellow are steps along the way which you will need to solve for the Green cells.

Processor B runs App X					
		Run Time	**0.0002 secs**		
# of Cycles	**# of Instructions**	**Instr Type**	Cycles per Instructions	Pct of Instr Type	(you will need to know % of Instr Type)
	500,000	**Add/Move**	**1**	%	
	150,000	**Branch**	**2**	%	.30
	200,000	**Multiply**	**3**	%	
	150,000	**Floating Point**	**4**	%	
Total Cycles	Total # of Instructions				Average CPI
	Clock Rate (GHz)	Clock Rate (cycles / sec)			
	Cycle Time (Sec/Cyc)	Cycle Time (sec / cycle)			

14 4 points

Application A on Computer A	
CPU Time (secs)	20.00
Cycles	70.00E+9
Cycle Rate (cycles / sec)	3.50E+9
We have created a new chip and have tested Application A. It runs in 12 secs and requires an additional 20% in instruction cycles	
Application A on Computer B	
CPU Time (secs)	12.00
Cycles (enter whole number no decimals)	x 10^9
Cycle Rate (enter whole number no decimals)	x10^9

15 10 points

Compare Performance of Chip X to Chip Z

	Chip X	Chip Z
Instruction Cycles	27 000 000 000	
Cycle Rate (Cycles/Sec)	4.2 GHz	3.5 GHz
If the new chip design for Computer Z **decreases** instruction cycles by 20% and has a cycle time of 3.5 GHz,		
Calculate the run time on Computer X. Only include **two decimals.**		Sec
Calculate the total cycles on the new chip Z. **No commas or spaces**		
Calculate the run time on the new chip Z. Only include **two decimals.**		Secs
Which chip ran the application faster? (X or Z)		

16 6 points

Application A on Computer A	
CPU Time (secs)	12.00
Cycles	30.00E+9
Cycle Rate (cycles / sec)	GHz
We have created a new chip and have tested Application A. It runs in 8 secs and requires an additional 1/3 in instruction cycles	
Application A on Computer B	
CPU Time (secs)	8.00
Cycles	Giga Cycles
Cycle Time	GHz

17 16 points

Convert Clock Rate to Clock Cycle or Clock Cycles to Clock Rate
If you are given the Clock Rate: cycles / sec (2MHz, 4,000,000, 5×10^{6})
determine the Clock Cycle Time: sec / cycle (500×10^{-9}, 250×10^{-9}, 200×10^{-9})
respectively.
i.e. if answer is 800×10^{-9} your response will just be 800.00

Clock Rate	Clock Cycles	Exponent
12.00×10^{6}		
166.66×10^{6}		-9
MHz	200.00	-9
2,000,000.00		-9
1 MHz		
MHz	100.00	-9

18 10 points

Given the Cycles per Instruction Type. Determine The CPI per Instruction Type and the Average CPI. All answers should have TWO decimal places. Even if the second decimal is 0.
Examples 2.00, 1.50, 1.66, 0.10, 0.25

Instruction Type	Add	Store	Branch	FP	
Cyles per Instruction Type	1	2	4	4	
Instruction Type Mix	50%	20%	20%	10%	
CPI per Instruction Type					Average CPI