CSCI 4591: Computer Architecture

Spring 2021
Midterm Exam

Name: Kerry Gip

Score: / 124

This is a take home test. Please make a note of the following:

- This is open book / open notes exam.
- You can use a calculator.
- You must justify all your answers to get full credit
- You must show all your work to get full credit
- If you make any assumptions to solve a problem, clearly state your assumptions.
- Add additional sheets as needed to complete your work.

Due Date: March 14, 2021 @ 11:55 PM – You must submit the test before this deadline. I am not going to make any exception to this rule.

GOOD LUCK!

1) Multiple Choice Questions: Please select the best answer from the given list for the following statements/questions. **(1 point each)**

A) Each byte of a memory has a unique address.
   i) True
   ii) False

B) Replacing a processor in a computer with a faster processor will increase response time.
   i) True
   ii) False

C) ADD instruction uses __ instruction format.
   i) R
   ii) D
   iii) I
   iv) B

D) LEGv8 has 32 bit registers.
   i) True
   ii) False

E) The first register operand (left operand) for an R-type instruction is stored in bit position 0 to 4.
   i) True
   ii) False

F) ___ means that the number is too small to be represented and it may be reported as 0.
   i) Underflow
   ii) Overflow
   iii) Sideflow
   iv) Wideflow

G) One of the trade-offs of floating-point math is that many calculations produce results that are not exact and have to be rounded to the nearest value that the notation can represent.
   i) True
   ii) False

H) Compared with addition and subtraction, multiplication is a complex operation, whether performed in hardware or software.
   i) True
   ii) False

2) Please fill in the blanks in the following statements: **(1 point each)**

A) If we add two positive numbers and the result is a negative number then we have OVERFLOW

B) ASSEMBLY LINES IN AUTOMOBILE MANUFACTURING is an example of using "Performance via Pipelining."

C) ADDEQ R1, R2, R3 ARM instruction will be executed if Z flag is set.

D) LEGv8 has a 32 bit registers. note: 32 bit registers, 64 bits wide

E) LEGv8 uses 64 bit instructions

F) If we are using 8 bit one's complement number system then the value of 11111111_2 in decimal is: 0 FROM ONES COMPLEMENT = 00000000 .

G) If we are using IEEE floating point number then a binary number is normalized if it is in the form SCIENTIFIC NOTATION OR 1.XXXXXXXX 2^Y .

3) Match the situation with the closest analog of a great idea in computer architecture. **(4 points)**

Use abstraction to simplify design - C	A) A soccer player runs not to where the ball is, but to where the ball will be.
Make the common case fast - D	B) A customer talks to a phone agent. If there's a problem, he talks to the agent's supervisor.
Design for Moore's Law - A	C) A house architect first designs a house with 5 rooms, then designs room details like closets, windows, and flooring.
Hierarchy of memories - B	D) A college student rents an apartment closer to campus than to her favorite weekend beach spot.

moores law : circuit resources double every 2 years ish - anticipate where things will be
abstraction: hide lower level details
common case fast - enhance performance
hierarchy of memories - memory to be fast - smallest up top and fast at bottom

4) Perform the following conversion (all numbers are positive): **(5 points each)**

A) 1101011.1111 – Convert binary to decimal = 107.938

number	converted
1	2^6 = 64
1	2^5 = 32
0	2^4 = 0
1	2^3 = 8
0	2^2 = 0
1	2^1=2
1.	2^0 = 1
.1	2^-1 = 0.5
1	2^-2 = 0.25
1	2^-3 = 0.125
1	2^-4 = 0.0625

TOTAL = 107.938

B) 129 – Convert decimal to binary = 10000001

number	divided
129/2	64 R1
64/2	32 R0
32/2	16 R0
16/2	8 R0
8/2	4 R0
4/2	2 R0
2/2	1 R0
1/2	0 R1

TOTAL = 10000001

C) ABCD4 - Convert hexadecimal to decimal = 703700

number	converted
A =10	16^4 = 655360
B = 11	16^3 = 45056
C = 12	16^2 = 3072
D = 13	16^1 = 208
4 = 4	16^0 = 4

TOTAL = 703700

D) 1 1010 1101.101 – Convert binary to hexadecimal =1AD.A
= 0001 1010 1101.1010
= 1 A D . A

5) Complete the following table (assume 8-bit word):

(10 points)

Decimal	**Ones Complement**	**Twos Complement**
-45	1101 0010	1101 0011

TO BINARY = -1010 1101

6) Convert the following twos complement number into decimal number: **(5 points)**

1111 0000 = Flip 0000 1111 + 1 = 0001 0000 = 2^4 = -16

7) Express the following number in IEEE 32-bit floating point format: **(5 points)**

18 = binary 10010 = 10010 = 1.0010 x 10^4 127+4 = 131

0 10000011 00100000000000000000000

8) The following number uses the IEEE 32-bit floating-point format. What is the equivalent decimal value? **(5 points)**

0 10000001 11110000000000000000000

129-127 = 2

Fraction = .1111 = $(2^{-1} + 2^{-2} + 2^{-3} + 2^{-4}) = 1.9375 \times 2^{2} = 7.75$

9) Consider two different processors P1 and P2 executing the same instruction set. P1 has a 4 GHz clock rate and a CPI of 2. P2 has a 3 GHz clock rate and a CPI of 1. Which processor has the highest performance expressed in instructions per second? **(10 points)**

P1 - 4 GHz CPI 2 -> 4×10^{9} cycles/sec * ½ (instructions/cycle) = 2×10^{9} instructions/sec

P2- 3 GHz CPI 1 -> 3×10^{9} * 1 instruction/cycle = 3×10^{9} instruction/ sec

P2 has the highest performance

10) Assume for arithmetic, load/store, and branch instructions, a processor has CPIs of 1, 12, and 5, respectively. Also assume that a program requires the execution of 2.56E9 arithmetic instructions, 1.28E9 load/store instructions, and 256 million branch instructions. Assume that the processor has a 2 GHz clock frequency. **(5 points each)**

P1 = 2 GHz = 2×10^{9} cycles/sec

A(arithmetic) - CPI 1 2.56×10^{9}

B (load/store)- CPI 12 1.28×10^{9}

C (branch)- CPI 5 $.256 \times 10^{9}$

Total instructions = 4.096×10^{9} instructions

A) Calculate the number of clock cycles required to execute the program.

- A(arithmetic) - 1 cycle/instruction * 2.56×10^{9} instructions = 2.56×10^{9} cycles
- B (load/store)- 12cycle/instruction * 1.28×10^{9} instructions = 1.536×10^{10} cycles
- C (branch)- 5 cycle/instruction * $.256 \times 10^{9}$ instructions = 1.28×10^{9} cycles
- Total = A+ B + C = 1.92×10^{10} cycles

B) Calculate the global (average) CPI. = sum(CPI x instruction count) / total # instructions
Average for each

A(arithmetic) - 1.92×10^{10} cycless/2.56×10^{9} instructions = 7.5 CPI

B (load/store)- 1.92×10^{10} cycles/ 1.28×10^{9} instructions = 15 CPI
C (branch)- 1.92×10^{10} cycles/ $.256 \times 10^{9}$ instructions = 75 CPI
Total Average = 97.5/3 = 32.5

C) Calculate the CPU time for the execution of the program.
CPU execution time = total cycles/ GHz = 1.92×10^{10} cycles/ 2×10^{9} cycles/sec = 9.6 secs

11) Provide the instruction type and binary representation of the following LEGv8 instructions.

(5 points each)

A) ANDS R9, R10, R11
Instruction type = R

opcode	Rm(11)	shamt	Rn(10)	Rd(9)
111 0101 0000	01011	000000	01010	01001

B) LSL R12, R13, #2
Instruction type = R

opcode	Rm(2)	shamt	Rn(13)	Rd(12)
110 1001 1011	00010	000000	01101	01100

12) Provide the instruction type and assembly language instruction for the following binary value:
11111000010000000000000101101010 **(5 points)**

opcode (LDUR)
Instruction type D

LDUR	Address(9 bits)	op2(2)	Rn(5)	Rt(5)
11111000010	000000000	00	01011	01010
	0	0	11	10

LDUR X10, [X11, #0]

13) You have been hired by ACME Corporation which makes 16-bit microcontroller for embedded system. You have been assigned the task of coming up with a way to represent a floating point number using a system similar to IEEE floating point standard. Note that IEEE floating point standard is defined for 32 bit and 64 bit systems. You decided to use the following fields to store a floating point number using 16 bit word:

1 bit is used for sign

5 bits are used for storing exponent and

10 bits are used for storing fractions.

A) What will be an optimum value for the bias so that you can store about equal number of negative and positive exponents. You must show your work and provide justification for your answer to get full credit. **(5 points)**

//https://en.wikipedia.org/wiki/Exponent_bias

To calculate bias, $2^{k-1} - 1$ where k is the number of bits in the exponent.

For a 5 bit exponent = 2^{5-1} -1 = 15

B) How will you represent 0 in the floating point number system that you designed. You must show your work and provide justification for your answer to get full credit. **(5 points)**

$0 = 0 \times 2^0 = 0$

Exponent: 0 + 15 = 15 -> binary 1111

Fraction = 0

Sign(1 bit)	exponent(5 bits)	fraction(10 bits)
0	01111	0000000000

C) What is the smallest positive number that you can store in the floating point number system that you designed. You must show your work and provide justification for your answer to get full credit. **(5 points)**

00000 and 11111 would be reserved and cannot be used in place of the exponent

00001 -> 1 would be the smallest exponent that can go in there

000000000 would be the smallest fraction that we can have which is 1

exponent : 1-15 = -14

So the smallest number that we can store is 1.0×2^{-14} .

sign	exponent	fraction
0	00001	0000000000

D) What is the largest positive number that you can store in the floating point number system that you designed. You must show your work and provide justification for your answer to get full credit. **(5 points)**

Same concept as the one above

00000 and 11111 would be reserved and cannot be used in place of the exponent

11110 -> 30 would be the largest exponent that can go in there

1111111111 would be the largest fraction that I can have and is approximate 2

exponent : 30-15 = 15

So the largest number that we can store is 2×2^{15} .

sign	exponent	fraction
0	11110	1111111111