MID-SEM RUBRIC
CSE112 Computer Organization

Q1.

(a)

(i) 10 + (-8) ⇒ 5 Bits each required to add correctly and represent -8

⇒ 01010 + (-01000)

⇒ Take 2's complement of -01000 => 11000

⇒ 01010 + (11000)

Addition:	0	1	0	1	0
+	1	1	0	0	0
EAC ⇒ 1	0	0	0	1	0

- Discard EAC in 2's complement addition.

- **<u>Answer</u>** ⇒ $\mathbf{0010_2 = 2_{10}}$

(ii) 5 + (-11) ⇒ 5 bits each required to add correctly and represent -11

⇒ 00101 + (-01011)

⇒ 00101 + 10101 (Taking 2's complement of 01011)

Addition:	0	0	1	0	1
+	1	0	1	0	1
	1	**1**	**0**	**1**	**0**

- **No EAC** is obtained in this case, also, **MSB = 1**, indicating it is a **negative number**, hence, find 2's complement form to find the magnitude and report the answer as -(magnitude obtained using 2's complement)
- 2's complement of $11010_2 = 00110_2$

- **<u>Answer</u>** ⇒ $\mathbf{-00110_2 = -6_{10}}$

(b)

RISC	**CISC**
One cycle per instruction	Can use multiple cycle per Instruction
Same Length Instruction	Variable Length Instruction
Less number of Instruction in ISA	Higher number of Instruction in ISA
Higher Size programs	Lower Size programs
Simpler Hardware Implementation	Complex Hardware implementation

Q2.

(a)

(i) 8 Bit 2's complement range ⇒ -2^{8-1} to $(2^{8-1} - 1)$ ⇒ -128 to 127

(ii) 8 Bit signed representation range ⇒ $-(2^{8-1} - 1)$ to $2^{8-1} - 1$ ⇒ -127 to 127

(b)

(i) 1. Exponent, E = 11111111 = All 1's
2. Mantissa, M ≠ 0 ⇒ Mantissa is not equal to zero
NaN occurs when the Exponent field is all ones with a zero sign bit along with a mantissa which is not 0.
The above two conditions in IEEE754 Single Precision Floating Point Notation correspond to **"NaN" => Not A Number**. It cannot be represented in decimal because it is not a number.

(ii) **Answer = 0.125_{10}**
Sign Bit, S = 0 ⇒ Positive Number
Exponent, E = 01111100_2 ⇒ 124_{10}
Subtracting the bias from E ⇒ 124-127 = -3 ⇒ Corresponds to 2^{-3} in standard binary form
Mantissa, M = 000…000 (All 0's) ⇒ 0_{10} ⇒ All 0's after decimal point(right to decimal point)
Normal Binary form becomes ⇒ 1 X 2^{-3}
Decimal Form becomes ⇒ 0.125_{10}

Q3.

```
Loop:  add R4, R0, R1        // In first iteration, 0+1 = 1 (3rd number of the series)
       add R0, R2, R1        // (R0 = R1)
       add R1, R2, R4        // (R1 = R4)
       subi R3, R3,1
       bz R3, Exit
       add R5, R2, R5
       bz R5, Loop
Exit:  add R5, R5, R2        // any dummy instruction or blank, both are fine
```

Q4.

(a)

Mem. location	Instruction	PC
0	beq R4, R0, 7	1
1	beq R1, R0, 7	7
2	add R1, R1, R1	-
3	sub R4, R4, R5	-
4	bne R4, R0, 2	-
5	add R10, R10, R1	-
6	jal R9, 8	-

7	add R10, R10, R0	8

In this part, R10 contains the value 0.

(b)

Mem. location	Instruction	PC (Iteration 1)	PC (Iteration 2)
0	beq R4, R0, 7	1	-
1	beq R1, R0, 7	2	-
2	add R1, R1, R1	3	3
3	sub R4, R4, R5	4	4
4	bne R4, R0, 2	2	5
5	add R10, R10, R1	-	6
6	jal R9, 8	-	8
7	add R10, R10, R0	-	

At the end R10 = 4.