INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, SRI CITY

TERM-II EXAMINATION – SPRING 2024

Computer Architecture

CSE:UG1(PC) **Date: 13-03-2024**

Duration: 90 Mins (08:45-10:15 AM) **Max. Marks: 30**

Roll No:_____ **Section:** _____

Instructions:

1. All questions are compulsory.
2. Write the answers legibly.
3. All sub-parts of a question should be written together.
4. **Attach the question paper with the answer sheet.**

Section-A (Multiple Objective Questions)

1	For an unsigned number x, the result of truncating it to k bits is equivalent to_____. a. computing x mod x^k b. computing x mod 2^k c. computing x mod 2^x d. All of the above	**[1 Mark]**
2	SUN XDR library is for_____. a. transferring data between machines b. performing unsign addition operations c. performing unsign multiplication operations d. All of the above	**[1 Mark]**
3	In single-precision IEEE floating point, Positive infinity is represented by the bit pattern $(7f800000)_{hex}$	**[1 Mark]**
4	In IEEE 754 floating-point representation, a signaling NaN (NAN) is characterized by which range of bit patterns? a. Between 7F800001 and 7FBFFFFF b. Between FF800001 and FFBFFFFF c. Between 80000001 and 80BFFFFF d. Between 3F800001 and 3FBFFFFF	**[1 Mark]**
5	Consider the following assembly instructions 1) mov (%eax, %eax, 4), %eax 2) lea (%eax, %eax, 4), %eax Which of the above accomplishes the following: %eax = 5 * %eax?	**[1 Mark]**

	a. Neither 1 nor 2. b. Only 1 c. Only 2. d. Both 1 and 2.	
6	Assuming the register %rbx contains the value 0xfaaafbbbfcccfddd, which instruction would cause the register %rdi to contain the value 0x00000000fcccfddd? a. movl %ebx, %rdi b. movslq %ebx, %rdi c. movzlq %ebx, %rdi d. lea %ebx, %rdi	**[1 Mark]**
7	If %rsp is 0xdeadbeefdeadd0d0. What is the value in %rsp after the following instruction executes? pushq %rbx a. 0xdeadbeefdeadd0d4 b. 0xdeadbeefdeadd0d8 c. 0xdeadbeefdeadd0cc d. 0xdeadbeefdeadd0c8	**[1 Mark]**
8	When does the overflow flag set in the given instruction? **addq Src,Dest ↔ t = a+b** a. (a>0 && b>0 && t<0) && (a<0 && b<0 && t>=0) b. (a<0 && b>0 && t<0) || (a<0 && b<0 && t>=0) c. (a>0 && b>0 && t<0) || (a<0 && b<0 && t>=0) d. (a<0 && b>0 && t<0) || (a>0 && b<0 && t>=0)	**[1 Mark]**
9	_____ is used to store the status information about the most recent arithmetic or logical operation. a. Program Counter b. ALU c. Condition Codes d. None of the mentioned	**[1 Mark]**
10	What is the difference between the %rbx and the %ebx register on an x86 64 machine? a. nothing, they are the same register b. %ebx refers to only the low order 32 bits of the %rbx register c. they are totally different registers d. %ebx refers to only the high order 32 bits of the %rbx register	**[1 Mark]**

Section-B (Descriptive Questions)

1 — **[5 Marks]**

Explain the modular addition forms an Abelian group with proper mathematical expressions.

- **Closed** under addition

  $0 \le \text{UAdd}_w(u, v) \le 2^w - 1$

- **Commutative**

  $\text{UAdd}_w(u, v) = \text{UAdd}_w(v, u)$

- **Associative**

  $\text{UAdd}_w(t, \text{UAdd}_w(u, v)) = \text{UAdd}_w(\text{UAdd}_w(t, u), v)$

- **0** is additive identity

  $\text{UAdd}_w(u, 0) = u$

- Every element has additive **inverse**
  - Let $\text{UComp}_w(u) = 2^w - u$

    $\text{UAdd}_w(u, \text{UComp}_w(u)) = 0$

2 — **[3 Marks]**

What is the IEEE 754 single-precision floating-point representation for the decimal number 85.125 and discuss the steps involved in its conversion?

- Normalize the binary representation:

  Combined integer and fractional parts: 1010101.001.

- Shift the binary point until there is only one non-zero digit to the left of the binary point: 1.010101001 * 2^6.
- Determine the sign, exponent, and fraction fields:
  - **The sign bit is 0 (positive).**
  - The exponent is bias + 6 = 127 + 6 = **133 in binary, which is 10000101.**
  - The fraction is the binary digits after the binary point, excluding the leading 1: 010101001. **Mantissa: 01010100100000000000000**
- Therefore, the complete 32-bit representation is:

  0 10000101 01010100100000000000000

[2 Marks]

- **C**onsider a five-bit floating representation based on the IEEE floating point format with 1 sign bit, two exponent bits, and 2 significant bits. Find the decimal value for the following?

01000	**1.00**

	01110 | **NaN** 1. Sign bit: 0 (positive). Exponent bits: 10. Bias: 2^(2-1) - 1 = 2^1 - 1 = 1. Exponent = 2 (binary) - 1 = 1 (decimal). Significand: 00. (−1)^0 ×2^1−1 ×1.00= **1×2^0 ×1.00=1×1×1.00=1.00** 2. Let's reanalyze the given number 01110: Sign bit: 0 (positive), exponent bits: 11. **Since all exponent bits are set to 1, the exponent is considered as a special case. Significand: 10. In IEEE 754, when all exponent bits are set to 1 and the significand is non-zero, it represents NaN.** So, the given five-bit floating-point representation 01110 represents NaN (Not a Number).	

3	Consider the following C code and translate it into corresponding assembly code.
	long arith(long x, long y, long z)
	{
	long t1 = x ^ y;
	long t2 = z * 48;
	long t3 = t1 & 0x0F0F0F0F;
	long t4 = t2 - t3;
	return t4;
	}
	Sol:
	1 arith:
	2 xorq %rsi, %rdi // t1 = x ^ y **[1 M]**
	3 leaq (%rdx,%rdx,2), %rax // 3*z **[1 M]**
	4 salq $4, %rax // t2 = 16 * (3*z) = 48*z **[1 M]**
	5 andl $252645135 or $0x0F0F0F0F , %edi //t3 = t1 & 0x0F0F0F0F **[1 M]**
	6 subq %rdi, %rax return t2 - t3 **[1 M]**
	7 ret
		[5 Marks]

4	For the given C code, fill-in the blank spaces in the corresponding assembly code. **C Code**: long absdiff (long a, long b) { **Assembly Code**: absdiff: cmpq __%rsi, %rdi_____ **1 Mark** jge_____.L4_____ // Jump instruction **1 Mark** **1 Mark**	**[5 Marks]**

	long result;
	if (a < b)
	result = b-a;
	else
	result = a-b;
	return result;
	}
		movq _%rsi, %rax_____
		subq __%rdi, %rax_____
		ret
		.L4
		movq__%rdi, %rax_____
		subq___%rsi, %rax_____
		ret
			1 Marks
			
			0.5 Mark
			0.5 Mark