Name___________________

Midterm Exam (B)

CSE222: Computer Architecture & Organization

(March 2020, SCCC)

Part I (40): Write answers in “CSE148_MT_AnswerSheet.docx”. Submit that file.

1. (3) (a) Give a brief description about how a machine instruction is executed in Von Neumann Architecture machine

   (3) (b) What is **big-endian** and what is **little-endian**? Given the following number **0xCDAB3120**, fill in the following table (see table in “answer-sheet” file) to indicate how this number is saved in memory as big- and little-endian:

2. (12) Convert numbers among different number systems:

   (**3517**)$_{10}$ = (   )$_2$ = (   )$_{16}$

   (   )$_{10}$ = (   )$_2$ = (**CD3**)$_{16}$

   (428)$_{10}$ = (   )$_{11}$

   (   )$_{10}$ = (**428**)$_{11}$

3. (7) Convert numbers:

   (a) **10101101** is 8-bit two’s complement number. Convert it to decimal

   (b) **100101** is a 6-bit two’s complement number. Expand it to 8-bit two’s complement number

   (c) Convert the following 2 decimal numbers to **8-bit** two’s complement binary numbers then perform addition. Write the result as a binary number

   (-51) + **(29)** = (   )$_2$ + (   )$_2$ = (   )$_2$

4. (10) **-53.34375** is a decimal number, write this number in the formats as list below:
   (You should write the results as hexadecimal number)

   (1) 16-bit fixed-point sign/magnitude format with 8 integer bits and 8 fraction bits;

   (2) 16-bit fixed-point 2’s complement format with 8 integer bits and 8 fraction bits;

   (3) Single precision (32-bit) IEEE 754 floating-point format

5. (5) Give brief description of the following instructions:

   (1) lw $s0, 4($a0)

   (2) beq $t0, $t1, addr

(3) slt $t2, $t0, $t1

(4) srlv $s0, $s1, $s2

(5) jr $ra

Part II: MIPS programming: Create one MIPS program file (.asm) for *each* question. Submit individual files, NOT TO archive these files.

1. (**10**) Write MIPS program:
   (a) Display message "Enter an integer number: ";
   (b) Input an integer number;
   (c) Use **2 methods** to check if this number is the multiples of 4. Display message "this number [is | is not] the multiples of 4";

2. (**30**) Write MIPS program:
   (a) Define an integer array of size 16, initialize array with random numbers in range [10, 100];
   (b) Display array
   (c) Define a function "**isOdd**" to check if an integer number is odd or not, return result
   (d) Define a function "**parseArray**" to analyze data in an integer array, this function will:
      (i) Count how many elements are even, how many elements are odd in array (*call function "**isOdd**" to check the element is odd or not*)
      (ii) Return above 2 count numbers
   (e) Call function "**parseArray**", pass array defined in (a) to the function, and display results returned from this function.

3. (**20**) Below is a piece of Java code. Write MIPS program to implement the same function:

```java
int sum = 0;
int count = 0;
while (sum < 50) {
     sum += count;
     count += 1;
}
int avg = sum / count;
System.out.printf("sum=%d\n", sum);
System.out.printf("count=%d\n", count);
System.out.printf("avg=%d\n", avg);
```