METRO STATE
UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 5 - Chapter 4 Part 2 - 10 points
Due Date: 6/14/2023

Name:

Note: Please post your homework to ICS232 D2L on or before the due date.

Chapter 4 – MARIE: An Introduction to a Simple Computer

Essential Terms and Concepts

31. How does a microprogram operation differ from a regular assembly language instruction?

- regular assembly language generates the control signals needed for the processor, while the microprogram generates the control signals with the help of micro instructions stored in control memory

39. Compare CISC machines to RISC machines.

- RISC: executes one instructions per clock cycle
- CISC: each instruction performs many actions that it takes several clock cycles to complete

Exercises

34. Write the following code segment in MARIE assembly language. (Hint: Turn the for loop into a while loop):

```
Sum = 0;
for X = 1 to 10 do
  Sum = Sum + X;
```

```
Loop,  Load   Sum
       Add    X
       Store  Sum
       Load   X
       Add    One
       Store  X
       Subs   Ten
```

```
        Skipcond        100
        Jumo    Loop
        Halt
X,      Hex     1
Sum,    Hex     0
Ten,    Dec     10
One,    Hex     1
```

39. MARIE saves the return address for a subroutine in memory, at a location designated by the jump-and-store instruction. In some architectures, this address is stored in a register, and in many it is stored on a stack. Which of these methods would best handle recursion?
Explain your answer. (Hint: Recursion implies many subroutine calls.)
- Stack, because stacks are accommodating to the needs for multiple calls to the subroutine. If there were only one register needed, multiple calls to the subroutine from within the subroutine wouldn't be possible.

X1. Write a C program that has two functions named `findMax` and `countOdd`. Each function will take two arguments, an integer array and the number of elements in the array. `findMax` returns the largest element. `countOdd` returns the count of the number of odd elements in the array. The main function should call `findMax` and `countOdd` twice with different arrays and then print out the results of calling the functions. The two arrays used to test the program should be:

```
static int array1[] = {1, -1, 100, 32, 64, -97};
static int array2[] = {-100, 1, -10, 50, -40, 98, 110};
```

Submit the C code and the results of executing the program.

Prepare for next class by reading Chapter 5 – A Closer Look at Instruction Set Architectures.

Continue working on Project 1

METRO STATE
UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 5 - Chapter 4 Part 2 - 10 points
Due Date: 6/14/2023

Continue working on Your Group Project