United International University
Department of Computer Science and Engineering

CSE-3313: Computer Architecture

Final Examination: Fall 2023

Total Marks: 40 Time: 2 hours

Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules.

Answer all questions. Numbers to the right of the questions denote their marks.

1. (a) You have to modify the block diagram for the single-cycle data path so that it can execute the following instruction **'sa'**. Also, write the control unit values for this instruction. **sa $s0, $s1, 2000**. This instruction will read the values of the two source registers(**rs, rt**) and checks if the first source register is less than the second source register. If true then it will jump to the given address(**C/A**). The machine code format for the instruction is given below table-1, where rs, rt will contain the index numbers of $s1 and $s2 registers, respectively. [5]

op	rs	rt	C/A
6 bits	5 bits	5 bits	16 bits

Table 1: machine code format of instruction **sa**

(b) You have to modify the block diagram for the single-cycle data path and write the control unit values for this instruction **'sb'**. This is an I-type instruction. The job of the given instruction is to read the value of the source registers (**$s0, $s1**) and multiply each with 8 and then add the values.Then it will store the result in the destination register **$t0**. [5]

sb $t0, $s0, $s1.

2. Consider a processor that goes through the following six stages while executing an instruction

Instruction Fetch (IF)	120
Instruction Decode(ID)	200
Execution(EXE)	150
Memory Read(MEMR)	150
Memory Write(MEMW)	180
Write Back(WB)	250

Table 2: Six stages pipeline

(a) **Find** out the total time for the given code below in a single cycle method. Assume that, 20($$s2$) has 5 as an initial value then also find the final value of 20($$s2$). [3]

```
1            addi $s0,$zero,10
2            add $s1,$zero,$zero
3            LOOP:
4            slt $t0,$s0,s1
5            bne $t0,$zero,EXIT
6            lw $t1,20($s2)
7            add $t2,$s1,$t1
8            addi $s1,$s1,2
9            j LOOP
10           EXIT:
11           sw $t2,20($s2)
```

(b) Using the pipeline method please find out the total time using a timing diagram. And if possible reduce the number of cycles for the below given code. [5]

```
addi $s0, $zero,1000
sw $t0,32($s0)
lw $t1,32($s0)
add $t6,$t0,$t1
and $t3,$s0,st1
sub $s4,$s4, $zero
```

(c) Now for the code snippet given in 2b please find out an optimal solution(By re-ordering the code) where stalls will be minimum. Find out the total time. No need to show a timing diagram. [5]

(d) When does a structural hazard occur? Explain the structural hazard with a diagram. [2]

3. A CPU possesses a physical memory of 1KB, which is mapped into a direct-mapped cache. Suppose, at time T, the cache exhibits the status provided in the table-3.

Index	V	D	Tag	Data			
				word[0]	word[1]	word[2]	word[3]
0	0	0	?	-	-	-	-
1	0	0	?	-	-	-	-
2	0	1	10	word[0]	word[1]	word[2]	word[3]
3	0	0	?	-	-	-	-
4	0	0	?	-	-	-	-
5	0	0	?	-	-	-	-
6	0	0	?	-	-	-	-
7	0	0	?	-	-	-	-
8	0	0	00	word[0]	word[1]	word[2]	word[3]
9	0	0	?	-	-	-	-
10	0	0	?	-	-	-	-
11	0	0	?	-	-	-	-
12	0	0	?	-	-	-	-
13	0	0	?	-	-	-	-
14	0	0	?	-	-	-	-
15	0	0	?	-	-	-	-

Table 3: Direct mapped table

(a) Calculate the actual size of the direct mapped cache. [2]

(b) If the CPU generates physical address requests sequentially as follows: 495, 130, 44, 105, 25, 499, 1012, 103, 107, 1014. Determine the cache miss ratio. Memory is byte addressable. [8]

(c) If the block size is doubled compared to the previous block size, and the number of cache lines remains unchanged, assuming the cache is initially empty, determine the cache hit ratio. Consider an array A with dimensions 8x8, where each element requires one word. Assume that the variables i, j, and sum are in registers, and the array A is in main memory initially. [5]

```
for(i=0;i<8;i++){
    for(j=0;j<8;j++){
        if((i+j)%2){
            sum += A[j][i];
        }
        else{
            sum += A[i][j];
        }
    }
}
```