End Sem
CSE 112 Computer Organization

INSTRUCTIONS:

Total Marks = 100
Time Duration = 120 mins solving + 15 mins uploading

1. **The duration of the exam is 120 mins, and 15 mins for scanning and uploading the solutions. No further extension of time will be given regarding this. Any late submission will be awarded 0 marks.**
2. **The question paper will be uploaded in google classroom. Do not forget to turn it in. Solutions submitted by any other means (email etc.) won't be considered for evaluation. Please upload your submission only on the classroom page of the section you are enrolled in. If you upload to the wrong section's page, you will be awarded 0 marks.**
3. **Students are required to switch on their cameras and mute themselves. Make sure you are sitting in a well-lit room so that we are able to see your faces clearly. If you are not clearly visible, you will be awarded 0 marks.**
4. **The answers should be in your own handwriting and submission should be in PDF format only.**
5. **Write any assumption clearly, if any. Needless to say, only reasonable assumptions will be considered if any ambiguity is found in the question.**
6. **During the exam, if you have any queries, write them in the meet chatbox. It will be taken into notice by us. Don't unnecessarily unmute your mic for it creates a disturbance to others.**
7. **Calculators are NOT allowed during exam time. ONLY use pen and paper for writing the exam.**
8. **Students need to be present and visible for the whole exam duration (till the end of solution uploading time) even if they upload the solution before time.**
9. **NAMING CONVENTION - <Name>_<Roll number>.pdf.
   Example Abc_Def_2020123.pdf**
10. **Show your calculations and justifications in each question.**

GOOD LUCK !!

Part A: Short Answers (25 marks)

Q1. (2+2+1 = 5 marks)
Two processors implemented on 2 different ISAs viz. **ISA A and ISA B** run the same benchmark.

Specifications	Processor A	Processor B
ISA	A	B
IPC (Instructions per cycle)	5	6
Clock rate	1000MHz	200MHz

Based on the specifications given above, answer the following questions:

a) What is the performance in Millions of Instructions per Second (MIPS) of the processor implementing ISA A? Show your calculation.

b) What is the performance in MIPS of the processor implementing ISA B? Show your calculation.

c) Which processor performs better on the given benchmark: A or B or Don't Know. Briefly explain your answer.

Q2. (4x2 = 8 marks)

In the program given below, functions foo() calls bar(). We have given you the pseudocode for the foo and bar. The ISA only contains 4 registers R0, R1, R2, R3. The given code follows an unknown caller callee convention. Assume that **every register is either caller saved or callee saved** and answer the following questions

a) Is R0 caller saved, or callee saved, or it cannot be determined from the given information? Justify your answer.

b) Is R1 caller saved, or callee saved, or it cannot be determined from the given information? Justify your answer.

c) Is R2 caller saved, or callee saved, or it cannot be determined from the given information? Justify your answer.

d) Is R3 caller saved, or callee saved, or it cannot be determined from the given information? Justify your answer.

```
bar()
{
        // modify register R1
        // return back to caller
}
foo()
{
        // push R0 to stack
        // modify registers R0,R3
        // push R3 on stack
        // call bar();
        // restore(pop) R3 from stack
        // modify R0 and R3 register
        // restore(pop) R0 from stack
        // return back to caller
}
```

Q3. (3x1 = 3 marks)

For a memory with a 5bit wide address bus and a 32bit wide data bus, answer the following questions:

a) How many distinct addresses are there in the memory?

b) How many bytes of data does each memory address store?

c) What is the total size of the memory in bytes?

Q4. (2x2 = 4 marks)

Consider a 2-way set associative cache with the following parameters:

- Size: 4096 bytes
- Cache line size: 16 bytes
- Number of ways: 2

Answer the following questions:

a) How many cache lines are there in the cache?

b) How many sets are there in the cache?

Q5. (2+2+1 = 5 marks)

Consider 2 configurations of a cache-based memory system:

Parameter	System A	System B
The probability that memory access hits in the cache. Also known as hit rate.	90%	95%
Cycles required to fetch the data if the access hits in the cache.	10	15
Cycles required to fetch the data if the access misses in the cache.	1000	1000

a) What is the average memory access latency of System A?
b) What is the average memory access latency of System B?
c) Which system would you prefer on your computer? Why?

Part B: Long Answers (75 marks)

For all the questions in **Part B** in this exam, use the following ISA description.
ISA Description:
The ISA uses

Add	Performs reg1 = reg2 + reg3	`add reg1 reg2 reg3`
Sub	Performs reg1 = reg2 - reg3	`sub reg1 reg2 reg3`
Mov Imm	Performs reg1 = Imm	`mov reg1 $Imm`
Mov	Performs reg1 = reg2	`mov reg1 reg2`
Branch if equal	Branch to addr if reg1 = reg2	`beq reg1 reg2 addr`

Apart from the above instructions, the assembler and the operating system support the following subroutines:

Name	**Semantics**	**Syntax**
Input	Reads immediate data from user into reg	`in reg`
Output	Prints string on the console	`out “str”`

The ISA only contains 8 registers r0-r7.

Q1. (3+6+6 = 15 marks)
Suppose you have a 32-bit processor that has a byte-addressable memory. This processor has a 512-byte direct-mapped cache with 4-byte cache lines.

a) What is the total number of cache lines in this cache?
b) How many bits are required for the tag, index (also known as set index), cache-line offset (also known as data index)?
c) Mark which segments of the address denote tag, index (also known as set index), cache-line offset (also known as data index). Mention the exact bit index ranges in the address for the tag, index (also known as set index), cache-line offset (also known as data index).

Q2. **(15 + 12 + 2 + 6 = 35 marks)**

Consider the following assembly program.

Assembly Program:

```
1     mov r1 r3
2     mov r2 r3
3     beq r1 r2 label1
4     mov r3 r1
5     add r4 r5 r6
6     label1: beq r3 r4 label2
7     mov r6 r1
8     add r4 r5 r6
9     add r7 r5 r1
10    label2: add r1 r4 r5
```

Initial values of registers are as follows:

`r0 = 0, r1 = 1, r2 = 2, r3 = 3, r4 = 4, r5 = 5, r6 = 6, r7 = 7`

a. Draw the pipeline diagram for the assembly program. Assume a 5 stage pipeline with Fetch, Decode, eXecute, Memory, and Writeback stages. Assume that register reads happen in the eXecute stage.

b. Mention any 2 potential dependencies of each type (WAR, RAW, and WAW) in the code. Mention their type. Mention the instruction number which is involved in the dependency. Also, mention the register on which the dependency is.

c. Mention the number of total cycles required to execute the program.

d. Write the values of each registers **r1, r2, r4, r6, and r7** after the cycle. Write your answer in the form of the table given below.

Cycle	r1	r2	r4	r6	r7

Q3. (5+20 = 25 marks)

a) The following pseudocode prints a triangle pattern over a dotted grid. The triangle is drawn using asterisk symbols. The dotted grid is drawn using full stops. The program takes the number “n” as an input from the user. It then prints a triangle pattern of height n. The given pseudocode is partially complete. Fill in the blanks with the correct expressions to complete the pseudocode. An example of the triangle pattern is given below for n = 5.
Note that printing “\n” moves the cursor to the next line.

Example: Triangle pattern for n = 5:

```
....*....
...***...
..*****..
.*******.
*********
```

Pseudocode:

```
n = input()
k = 1
for(i = _(a)_; i > 0; i--)
{
    for(j = 0; j < _(b)_; j++)
        print (".")
    for(j = _(c)_; j < k; j++)
        print("*")
    for(j = 0; j < _(d)_; j++)
        print(".")
    print("\n")
    k = k + _(e)
}
```

b) Write an assembly program to print the triangle pattern. Your assembly program should follow the given pseudocode.