B.E. ELECTRONICS AND TELE-COMMUNICATION ENGINEERING
THIRD YEAR FIRST SEMESTER EXAM - 2024

COMPUTER ORGANIZATION AND ARCHITECTURE

Time: 3 hours **Full Marks: 100**

Answer questions from all five modules

MODULE 1 (Based on CO 1) **(20)**

1. a) State and explain the CPU performance equation. Discuss any underlying assumption you are making while writing this equation. **3+1**

   Assume for arithmetic, load/store and branch instructions, a processor has CPIs of 1, 12 and 5 respectively. Further assume that on a single processor a program requires the execution of 2.5E9 arithmetic instructions, 1.28E9 load/store instructions and 256 E6 branch instructions. Assume that each processor has a 2 GHz clock frequency. Assume that as the program is parallelized to run over multiple cores, the number of arithmetic and load/store instructions per processor is divided by 0.7 x p (where p is the number of processors) but the number of branch instructions per process remains the same. **10**

   i) Find the total execution time for this program on 1 and 4 processors.

   ii) To what the CPI of the load/store instructions be reduced in order for a single processor to match the performance of four processors using the original CPI values?

   b) What is SPECratio?

   c) The results of the SPEC CPU2006 bzip2 benchmark running on AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s and a reference time of 9650 s. Find the increase in CPU time and the change in SPECratio if the number of instructions of the benchmark is increased by 10% and the CPI is increased by 5%. **1** **5**

MODULE 2 (Based on CO 2) **(20)**

2. a) Explain with proper examples how the design principles of "*Simplicity favours regularity*" and "*Smaller is faster*" are followed in MIPS programming. **4**

   b) Differentiate between data transfer and branching type instructions in MIPS with proper examples. **4**

   c) Translate the following C code into MIPS: **5+1**

```
i = 5;
do {
    B += 2;
    i = i – 1;
} while (i > 0)
```

Assume the variables *i,* and *B* to be in the registers $t1 and $s2 respectively. How many MIPS instructions will be executed for the above initialization of *i*?

d) Translate the following MIPS code into C. Assume that the variables *a*, *b*, *i*, and *j* are assigned to registers $s0, $s1, $t0, and $t1 respectively. Assume that the base address of the array D is in the register $s2. **6**

```
        addi $t0, $0, 0
        beq $0, $0, TEST1
LOOP1:  addi $t1, $0, 0
        beq $0, $0, TEST2
LOOP2:  add $t3, $t0, $t1
        sll $t2, $t1, 4
        add $t2, $t2, $s2
        sw $t3, ($t2)
        addi $t1, $t1, 1
TEST2:  slt $t2, $t1, $s1
        bne $t2, $0, LOOP2
        addi $t0, $t0, 1
TEST1:  slt $t2, $t0, $s0
        bne $t2, $0, LOOP1
```

OR

a) Briefly explain the different addressing modes used in the MIPS instruction set architecture. Give one example instruction for each mode. **5+3**

b) Translate the following C code, in MIPS: **6**

```
for (i=0; i<100; i++)
{
    result += MemArray[i];
}
```

Assume $s0 holds the base address of the integer array MemArray (consisting of 32-bit words), $s1 holds the integer *i*, and $s2 holds the integer *result*.

c) Implement the following recursive C function, which computes the sum of the first *n* natural numbers, in MIPS: **6**

```
int RecSum (int n)
{
    if (n <= 1)
        return n;
    return n + RecSum(n-1) ;
}
```

Clearly state the resources you are using for this implementation.

MODULE 3 (Based on CO 3) **(20)**

3. a) How can you perform signed multiplication in MIPS? Consider multiplying the decimal numbers 9 and -3. **1**
   i) Design a simple (maybe un-optimized) hardware to realize the above task. **2**
   ii) Show a step-by-step calculation to achieve the above task using the hardware in i). **6**
   iii) How can you optimize the hardware in i) **2**
   b) Show with the help of a flowchart how you can add the decimal numbers 0.5 and -0.25. **6**
   c) What is subword parallelism? For what type of application, subword parallelism can be helpful? **2+1**

MODULE 4 (Based on CO 4) **(20)**

4. a) Differentiate between RISC and CISC processors with two proper examples. **4**
   b) Design a 32-bit MIPS ALU for implementing any two arithmetic operations and ay three logical operations. **5**
   c) Explain the role of ALU in implementing three different types of MIPS instructions. **3**
   d) What is a datapath? Show the steps in the operation of a datapath while implementing i) a typical R-type instruction and ii) a typical load-store instruction. **2+3+3**

MODULE 5 (Based on CO 5) **(20)**

5. a) Discuss the necessity of memory hierarchy stating the contrasting requirements. How the hierarchy is designed? **2+2**
   b) Define three performance measures related to the design of cache memory. **6**
   c) Suppose we have the following system: a processor with CPI 1.0 without any stall, a clock rate of 5 GHz., two levels of cache, miss rate of primary cache as 2%, miss rate to main memory as 0.5%, main memory access time as 200 ns and the secondary cache access time as 5 ns. Justify the inclusion of the secondary level cache. Further, analyze the performances (with and without secondary level cache) if the miss rate gets doubled. **10**

OR

   a) Explain how the principles of locality can be helpful for the memory design. **4**
   b) Discuss the design principles of three different types of cache - direct mapped, n-way set associative and fully set-associative. **6**
   c) Consider the following sequence of block addresses: 0, 8, 4, 4, 0, 6, 8, 8, 4, 6. You are given a set of three small caches each consisting of four one-word blocks of the types stated in b). Assume n = 2. As a computer architect, design the best appropriate cache for the above sequence. Clearly, justify your design. You may consider different block replacement policies in your analysis. **10**