E/L/TX

Reg. No:

VIT
Vellore Institute of Technology

# Final Assessment Test – November 2024

Course: **CSI3021 - Advanced Computer Architecture**

Class NBR(s): **1933/1951/1957/3605** Slot: **E1+TE1**

Time: **Three Hours** Max. Marks: **100**

- **KEEPING MOBILE PHONE/ANY ELECTRONIC GADGETS, EVEN IN 'OFF' POSITION IS TREATED AS EXAM MALPRACTICE**
- **DON'T WRITE ANYTHING ON THE QUESTION PAPER**

**Answer ALL Questions**

**(10 X 10 = 100 Marks)**

1. a) Considering the basic functional blocks in a computer, which basic architecture is best suited for real-world applications? Discuss the advantages and disadvantages of each architecture. **[4]**

   b) Consider a processor with a clock rate of 200 Mhz running a given set of instructions **[6]**

| Instruction Type | Instruction Count (Millions) | Cycle per Instruction |
|---|---|---|
| Machine A Fetch-I/O | 10 | 1 |
| Arithmetic and logic | 8 | 2 |
| Store | 2 | 4 |
| Branch | 4 | 3 |

   Calculate the Effective CPI, MIPS rate and Execution time.

2. Explain the concepts of dynamic branch prediction. Use the following code for implementing 1 bit branch predictor, and 2 bit branch predictor. Discuss how many cycle stalls are occurring in each prediction methods.

```
int a=0;
while(a<5) {
//branch instruction, condition either true or false
 if(a%2==0)
{.....} a++;
}
```

3. Analyse the given code to determine the number of dependencies present. Identify appropriate methods for resolving these dependencies and discuss how additional hardware can be utilized to buffer results during out-of-order execution of instructions. Explain the process with neat organizational diagram.

```
DIV.D F0,F2,F4
ADD.D F6,F0,F8
S.D   F6,0(R1)
SUB.D F8,F10,F14
MUL.D F6,F10,F8
```

4. Define Loop-carried dependencies. Find the loop dependencies in the given code. How the loop level parallelism will help to solve the issues and discuss the solution associated with GCD technique and assume all constant coefficients value are equal to 1.

```
for (i=0; i<100; i=i+1) {
        A[i+1] = A[i] + C[i]; /* S1 */
        B[i+1] = B[i] + A[i+1]; /* S2 */}
```

5. Discuss how the multithreaded critical section problem can be resolved using semaphores. Explain how to address the issues of busy-waiting and blocking semantics in the critical section. Additionally, relate these issues to the any one classical synchronization problem.

6. a) Explain the term concurrency, parrallisim, Fine and course granularity.

b) In an organization need to enhance the processor used for Web serving. The new processor is 10 times faster on computation in the Web serving application than the original processor. Assuming that the original processor is busy with computation 40% of the time and is waiting for I/O 60% of the time, Apply the Amdahl's Law and check the overall speedup.

7. Explain the symmetric and asymmetric multiprocessor architecture and discuss the processor affinity issues in multiprocessor architecture.

8. Discuss the difference between homogenous and heterogeneous Multiprocessor systems and also explain any two shared memory interconnection structure with neat diagram.

9.a) Explain the detail about the shared memory system and distributed memory system architecture with neat diagram.

OR

9.b) What is cache coherence and why is it important in shared memory multiprocessors systems? Explain how coherence problems can be solved with protocols.

10.a) Explain in detail the OpenMP Work sharing constructs with suitable programming examples.

**OR**

10.b) How the race condition and Synchronization constructs can be solved by OpenMP directives. Illustrate with an example to solve the synchronization problem.

⇔⇔⇔ **E/L/TX** ⇔⇔⇔