V SEMESTER

Course Code	:	**CSPC51**
Course Title	:	**Computer Architecture**
Number of Credits	:	**3-1-0-4**
Pre-requisites (Course Code)	:	**CSPC34**
Course Type	:	**PC**

Course Objectives

- To understand the concept of advanced pipelining techniques
- To understand the current state of art in memory system design
- To know the working principle of I/O devices
- To understand the memory management techniques

Course Contents

UNIT I
Introduction - Classes of computers - Defining Computer Architecture - Trends in Technology - Trends in Power and Energy in Integrated Circuits - Trends in Cost - Dependability - Measuring - Reporting and Summarizing Performance - Quantitative Principles of Computer Design.

UNIT II
Basic and Intermediate pipelining Concepts - The Major Hurdle of Pipelining - Pipeline Hazards - Pipelining Implementation - Implementation issues that makes Pipelining hard - Extending the MIPS Pipeline to Handle Multicycle Operations - The MIPS R4000 Pipeline.

UNIT III
Instruction-Level Parallelism: Concepts and Challenges - Basic Compiler Techniques for Exposing ILP - Reducing Branch Costs with Prediction - Overcoming Data Hazards with Dynamic Scheduling - Dynamic Scheduling - Hardware-Based Speculation - Exploiting ILP Using Multiple Issue and Static Scheduling - Exploiting ILP - Advanced Techniques for Instruction Delivery and Speculation - Studies of the Limitations of ILP.

UNIT IV
Vector Architecture - SIMD Instruction Set Extensions for Multimedia - Graphics Processing Units - Detecting and Enhancing Loop-Level Parallelism - Centralized Shared-Memory Architectures - Performance of Shared-Memory Multiprocessors - Distributed Shared Memory - Models of Memory Consistency - Multicore Processors and their Performance.

UNIT V
Review of Memory Hierarchy Design - Cache Performance - Basic Cache Optimizations - Virtual Memory - Protection and Examples of Virtual Memory - Advanced Optimizations of Cache Performance - Memory Technology and Optimizations - Protection: Virtual Memory and Virtual Machines - Crosscutting Issues: The Design of Memory Hierarchies - Case Studies / Lab Exercises.

Course Outcomes

Upon completion of the course, the students will be able to:

- Apply performance metrics to find the performance of systems
- Identify the program block that requires parallelism for any program
- Comprehend and differentiate various computer architectures and hardware
- Design algorithms for memory management techniques

Text Books

1. David. A. Patterson, John L. Hennessy, “Computer Architecture: A Quantitative approach”, Fifth Edition, Elsevier, 2012.