MULTICORE ARCHITECTURE AND PROGRAMMING (Effective from the academic year 2018 -2019) SEMESTER – VII			
Course Code	**18CS824**	**CIE Marks**	40
Number of Contact Hours/Week	3:0:0	**SEE Marks**	60
Total Number of Contact Hours	40	**Exam Hours**	03
CREDITS –3			

Course Learning Objectives: This course (18CS824) will enable students to:

- Define technologies of multicore architecture and performance measures
- Demonstrate problems related to multiprocessing
- Illustrate windows threading, posix threads, openmp programming
- Analyze the common problems in parallel programming

Module -1	**Contact Hours**
Introduction to Multi-core Architecture Motivation for Concurrency in software, Parallel Computing Platforms, Parallel Computing in Microprocessors, Differentiating Multi-core Architectures from Hyper- Threading Technology, Multi-threading on Single-Core versus Multi-Core Platforms Understanding Performance, Amdahl's Law, Growing Returns: Gustafson's Law. System Overview of Threading : Defining Threads, System View of Threads, Threading above the Operating System, Threads inside the OS, Threads inside the Hardware, What Happens When a Thread Is Created, Application Programming Models and Threading, Virtual Environment: VMs and Platforms, Runtime Virtualization, System Virtualization. **Textbook 1: Ch.1, 2** **RBT: L1, L2, L3**	08
Module -2	
Fundamental Concepts of Parallel Programming :Designing for Threads, Task Decomposition, Data Decomposition, Data Flow Decomposition, Implications of Different Decompositions, Challenges You'll Face, Parallel Programming Patterns, A Motivating Problem: Error Diffusion, Analysis of the Error Diffusion Algorithm, An Alternate Approach: Parallel Error Diffusion, Other Alternatives. Threading and Parallel Programming Constructs: Synchronization, Critical Sections, Deadlock, Synchronization Primitives, Semaphores, Locks, Condition Variables, Messages, Flow Control- based Concepts, Fence, Barrier, Implementation-dependent Threading Features **Textbook 1: Ch.3, 4** **RBT: L1, L2, L3**	08
Module – 3	
Threading APIs :ThreadingAPIs for Microsoft Windows, Win32/MFC Thread APIs, Threading APIs for Microsoft. NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signaling, Compilation and Linking. **Textbook 1: Ch.5** **RBT: L1, L2, L3**	08
Module-4	
OpenMP: A Portable Solution for Threading : Challenges in Threading a Loop, Loop-carried Dependence, Data-race Conditions, Managing Shared and Private Data, Loop Scheduling and Portioning, Effective Use of Reductions, Minimizing Threading Overhead, Work-sharing Sections, Performance-oriented Programming, Using Barrier and No wait, Interleaving Single-thread and Multi-thread Execution, Data Copy-in and Copy-out, Protecting Updates of Shared Variables, Intel Task queuing Extension to OpenMP, OpenMP Library Functions,	08

OpenMP Environment Variables, Compilation, Debugging, performance **Textbook 1: Ch.6** **RBT: L1, L2, L3**	
Module-5	
Solutions to Common Parallel Programming Problems : Too Many Threads, Data Races, Deadlocks, and Live Locks, Deadlock, Heavily Contended Locks, Priority Inversion, Solutions for Heavily Contended Locks, Non-blocking Algorithms, ABA Problem, Cache Line Ping-ponging, Memory Reclamation Problem, Recommendations, Thread-safe Functions and Libraries, Memory Issues, Bandwidth, Working in the Cache, Memory Contention, Cache-related Issues, False Sharing, Memory Consistency, Current IA-32 Architecture, Itanium Architecture, High-level Languages, Avoiding Pipeline Stalls on IA-32,Data Organization for High Performance. **Textbook 1: Ch.7** **RBT: L1, L2, L3**	08

Course Outcomes: The student will be able to :

- Identify the limitations of ILP and the need for multicore architectures
- Define fundamental concepts of parallel programming and its design issues
- Solve the issues related to multiprocessing and suggest solutions
- Make out the salient features of different multicore architectures and how they exploit parallelism
- Demonstrate the role of OpenMP and programming concept

Question Paper Pattern:

- The question paper will have ten questions.
- Each full Question consisting of 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Textbooks:

1. Multicore Programming , Increased Performance through Software Multi-threading by Shameem Akhter and Jason Roberts , Intel Press , 2006

Reference Books:

1. Yan Solihin, "Fundamentals of Parallel Multicore Architecture", 1st Edition, CRC Press/Taylor and Francis, 2015.
2. GerassimosBarlas, "Multicore and GPU Programming: An Integrated Approach Paperback", 1st Edition, Morgan Kaufmann, 2014.
3. Lyla B Das, "The x86 Microprocessors: 8086 to Pentium, Multicores, Atom and the 8051 Microcontroller: Architecture, Programming and Interfacing", 2nd Edition, Pearson Education India, 2014

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225