



Bapatla Engineering College :: Bapatla (Autonomous)

ADVANCED COMPUTER ARCHITECTURE Department Elective-I III B.Tech – V Semester (Code:18CSD11)			
Lectures :	4 Periods / Week	Continuous Internal Assessment :	50 Marks
Final Exam :	3 hours	Semester End Exam :	50 Marks
UNIT-I			16 Periods
<p>Parallel Computer Models: The state of computing, Classification of parallel computers, Multiprocessors and Multi computers, Multi-vector and SIMD computers.</p> <p>Program and network properties: Conditions of parallelism, Data and resource Dependencies, Hardware and Software parallelism, Program partitioning and scheduling, Grain Size and latency, Program flow mechanisms, Control flow versus data flow, Data flow Architecture, Demand driven mechanisms, Comparisons of flow mechanisms.</p> <p>System Interconnect Architectures: Network properties and routing, Static interconnection Networks, Dynamic interconnection Networks, Hierarchical bus systems, Crossbar switch and multiport memory, Multistage and combining network.</p>			
UNIT-II			16 Periods
<p>Principles of Scalable Performance: Performance Metrics and Measures: Parallelism Profile in Programs, Efficiency, Utilization and Quality, Standard Performance Measures, Speedup Performance Laws: Amdahl's law for fixed load, Gustafson's law for scaled problems, Memory Bounded Speedup Model.</p> <p>Pipelining: Linear pipeline processor, nonlinear pipeline processor, Instruction pipeline Design- Instruction Execution Phases, Mechanisms for instruction pipelining, Dynamic instruction scheduling, Branch Handling techniques, Arithmetic Pipeline Design: Computer Arithmetic principles, Static Arithmetic pipeline, Multifunctional arithmetic pipelines.</p>			
UNIT-III			16 Periods
<p>MULTI Processors: Multiprocessor System Interconnect: Hierarchical Bus Systems, Crossbar Switch and Multiport Memory, Multistage and Combining Networks, Cache Coherence and Synchronization Mechanisms: The Cache Coherence problem, Snoopy Bus Protocols, Directory Based Protocols, Hardware Synchronization Mechanisms, Message-passing Mechanism: Message Routing Schemes, Deadlock and Virtual Channels, Flow Control Strategies, Multicast Routing Algorithms.</p> <p>Scalable, Multithreaded and Dataflow Architectures: Latency-Hiding Techniques, Principles of Multithreading, Scalable and Multithreaded Architectures.</p>			
UNIT-IV			16 Periods
<p>Thread Based Parallelism: Introduction, Using the python threading model, How to define a Thread, How to determine a current Thread, How to use a thread in subclass, Thread</p>			



Bapatla Engineering College :: Bapatla (Autonomous)

Synchronization with Lock and RLock, Thread Synchronization with RLock, Thread Synchronization with Semaphores, Thread Synchronization with a Condition, Thread Synchronization with an Event, Using a with Statement, Thread Communication with a Queue, Evaluating the performance of Multithreaded applications.

Process Based Parallelism: Introduction, How to spawn a process, How to name a Process, How to run a Process in the background, How to kill a process, How to use a process in subclass, how to exchange objects between processes, How to synchronize the Processes, How to manage a state between Processes, How to use a Process pool, Using the mpi4py python module, Point-to-Point to Communications, Avoiding Deadlock problems, Collective communication using Broadcast, Collective Communication using a Scatter, Collective Communication using Gather, Collective Communication using Alltoall, The reduce operation, How to Optimize an Operation.

Text Book(s) :	<ol style="list-style-type: none">1. Kai Hwang, "Advanced Computer Architecture", TMH.2. "Python Parallel Programming cookbook", Giancarlo Zaccone, Packt Publishing.
References :	<ol style="list-style-type: none">1. D.A. Patterson and J.L.Hennessy, "Computer organization and Design", Morgan Kaufmann, 2nd Edition.2. V.Rajaram & C.S.R.Murthy, "Parallel Computer", PHI.3. Barry Wilkinson and Michael Allen, "Parallel Programming", Pearson Education.4. Parallel Programming with Python, Jan Palach, Packt Publishing