**MULTI-CORE ARCHITECTURES AND GPU PROGRAMMING L T P C**

**3 0 0 3**

**OBJECTIVES:**

**The student should be made to:**

* To understand the basics of multi-core architectures
* Understand the challenges in parallel and multi-core programming.
* Learn about CUDA programming paradigms, and solutions.

UNIT I MULTI-CORE PROCESSORS 9

Single core to Multi-core architectures – Homogeneous and Heterogeneous Multi-core Architectures - Intel Multicore Architectures-SIMD Systems: Vector Processor-MIMD systems :– Symmetric and Distributed Shared Memory Architectures –Interconnection Networks Case Studies – SUN CMP architecture – IBM Cell Architecture

UNIT II PARALLEL PROGRAM CHALLENGES 9

Performance – Scalability – Synchronization and data sharing – Data races – Synchronization primitives (mutexes, locks, semaphores, barriers) – deadlocks and livelocks – communication between threads (condition variables, signals, message queues and pipes).

UNIT III GPU ARCHITECTURES 9

Evolution of GPU architectures - Understanding Parallelism with GPU –Typical GPU Architecture -–CUDA Hardware Overview.

UNIT IV CUDA PROGRAMMING 9

Threads, Blocks, Grids, Warps, Scheduling- Memory Handling with CUDA: Shared Memory, Global Memory, Constant Memory and Texture Memory.

UNIT V PROGRAMMING ISSUES 9

Common Problems: CUDA Error Handling, Parallel Programming Issues, Synchronization, Algorithmic Issues, Finding and Avoiding Errors.

**TOTAL: 45 PERIODS**

**OUTCOMES:**

**At the end of the course, the student should be able to:**

* + 1. Point out the salient features of different multicore architectures and how they exploit parallelism.
    2. Understand the challenges in parallel and multi-threaded programming
    3. Understand the salient features of GPU Architectures.
    4. Write programs using CUDA.
    5. Identify the issues in CUDA programming and debug them

**TEXT BOOKS:**

1. John L. Hennessey and David A. Patterson, “Computer Architecture – A Quantitative Approach”, Morgan Kaufmann / Elsevier, 5th edition, 2012. (Unit 1 and 3)

2. Shane Cook, CUDA Programming: ―A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012.

2. Darryl Gove, “Multicore Application Programming for Windows, Linux, and Oracle Solaris”,Pearson, 2011 (unit 2)

**REFERENCES:**

1. SANDERS, J. KANDROT,E. CUDA by example: an introduction to general-purpose GPU programming, Addison-Wesley, 2011 978-0-13-138768-3

2. Nicholas Wilt, ―CUDA Handbook: A Comprehensive Guide to GPU Programming‖, Addison - Wesley, 2013.

3. Jason Sanders, Edward Kandrot, ―CUDA by Example: An Introduction to General Purpose GPU Programming‖, Addison - Wesley, 2010.