**Course Objective:** To learn the advanced concepts of Parallel and Distributed Computing and its implementation for assessment of understanding the course by the students

**Introduction:** Scope, issues, applications and challenges of Parallel and Distributed Computing


**Principles of Parallel Algorithm Design:** Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing.

**CUDA programming model:** Overview of CUDA, Isolating data to be used by parallelized code, API function to allocate memory on parallel computing device, to transfer data, Concepts of Threads, Blocks, Grids, Developing a kernel function to be executed by individual threads, Execution of kernel function by parallel threads, transferring data back to host processor with API function.

**Analytical Modeling of Parallel Programs:** Sources of Overhead in Parallel Programs, Performance Metrics for Parallel Systems, The Effect of Granularity on Performance, Scalability of Parallel Systems, Minimum Execution Time and Minimum Cost-Optimal Execution Time

**Dense Matrix Algorithms:** Matrix-Vector Multiplication, Matrix-Matrix Multiplication, Issues in Sorting on Parallel Computers, Bubble Sort and Variants, Quick Sort, Other Sorting Algorithms

**Graph Algorithms:** Minimum Spanning Tree: Prim's Algorithm, Single-Source Shortest Paths: Dijkstra's Algorithm, All-Pairs Shortest Paths, Transitive Closure, Connected Components, Algorithms for Sparse Graph

**Search Algorithms for Discrete Optimization Problems:** Sequential Search Algorithms, Parallel Depth-First Search, Parallel Best-First Search, Speedup Anomalies in Parallel Search Algorithms

**Laboratory Work:** To implement the algorithms with the help of CUDA programming using parallel and distributed programming techniques
Recommended Books: