SCHEME AND SYLLABUS

FOR

B.TECH MINOR DEGREE IN

COMPUTING AND INTELLIGENCE

Offered by the Department of Computer Engineering,
National Institute of Technology, Kurukshetra

w.e.f. 2024-25

VISION AND MISSION OF THE INSTITUTE

VISION

To be a role-model in technical education and research, responsive to global challenges.

MISSION

To impart technical education that develops innovative professionals and entrepreneurs and to undertake research that generates cutting-edge technologies and futuristic knowledge, focusing on the socio-economic needs.

VISION AND MISSION OF THE DEPARTMENT

VISION

To address societal needs and global industry challenges in the field of Computer & IT with state-of-art education & research.

MISSION

M-1: To create a platform for education, research and development by providing sound theoretical knowledge and practical skills in Computer Engineering and Information Technology.

M-2: To produce motivated professional technocrats capable of generating solutions for industry and society.

M-3: To develop the ability to work ethically at individual and team level and be responsive towards socio-economic needs.

VISION AND MISSION OF THE PROGRAM

VISION

To disseminate state-of-the-art education to develop competent professionals in Computer Engineering with capability to serve the global society**.**

MISSION

To educate and train manpower engaged in cutting-edge research by offering latest in the field of Computer Engineering for sustainable development of society.

Scheme for Minor Degree in Computing and Intelligence

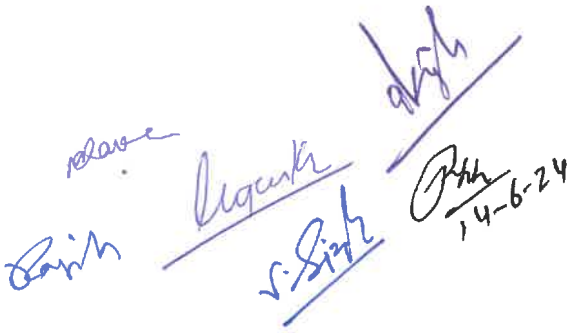
The number of courses proposed are:

1. Four Core Courses
2. Two Program Electives

Sr. No.	Course Type	Name of the Course	Course Type	L_T_P	Credits
1.	Program Core	Data Structures	Core	3 0 2	4
2.		Design and Analysis of Algorithms	Core	3 0 2	4
3.		Operating Systems	Core	3 0 2	4
4.		Database Management Systems	Core	3 0 2	4
5.	Program Elective	Computer Networks	Elective	3 0 2	4
6.		Software Engineering	Elective	3 0 2	4

Note: Since the courses offered are of integrated nature (Theory and Practical both). The credit assigned to each of it is four (4) which makes the total credits for five (5) courses to be twenty (20).

1. Students will not be permitted to opt a course, which has been studied as a **Minor Degree Course for Open Elective/Program Course in Major Degree** or vice-versa.
2. In case one/two Program Core course(s) (Sr. 1, 2) has/have been studied by a student in his/her major Degree, he/she will complete the requisite count of courses by studying Program Elective course(s), in lieu.



Course Code	**:**	**CIPC 200**
Course Title	**:**	**Data Structures**
Number of Credits	**:**	**4 and 3/0/2**
Prerequisites	**:**	**Problems Solving and Programming Skills**
Course Type	**:**	**OE**

Course Learning Objectives

1. This course introduces the concept of Data Structures used in various computer science applications
2. The students are introduced to understand and efficiently apply various data structures such as stacks, queues, linked lists, trees and graphs for solving various computing problems using C programming language.

Course Content

Unit 1: Pointers & File Handling:

Revision of Pointers and Dynamic Memory, Files and related operations.
Searching techniques: Linear and Binary, Sorting techniques: Selection, Bubble, Insertion, Merge sort, Quicksort.

Unit 2: Simple Data Structures

Arrays based Linear Data Structures: Array storage, sparse arrays; Transpose, addition, and multiplication of sparse matrices, Stacks and Queues and their applications, multiple stacks and queues in an array.

Unit 3: Linked Data Structures

Singly, Doubly & Circular Linked Lists; representation, operations and applications, linked stacks and queues. linked lists based polynomial addition.

Unit 4: Advanced Data Structures

Trees, Basic concepts and definitions of a tree and binary tree and associated terminology, Binary tree traversal techniques, some more operations on binary trees, Heaps and heapsort.

Reference Books:

1. E Horowitz and S. Sahni: Fundamentals of Data Structures in C, Second Edition, Universities Press, Hyderabad.
2. R.L. Kruse: Data Structures & Program Design in C, PHI.
3. D.F. Knuth: The art of Computer Programming Vol-1, Narosa Publications, 1985.
4. Byron S. Gottfried & J K Chhabra: Theory and Problems of Programming with C Language, Schaum's Outlines Series, TMH, 2005.

Course Outcomes

1. Develop skill to identify and determine the usage of various data structures, operations, associated algorithms and implement their applications.
2. Apply knowledge of pointers, memory allocation and string handling for solving programming problems.
3. Understand the concept of trees and graphs, their implementation and applications.
4. Able to implement standard algorithms for searching and sorting.
5. Analyze efficiency of different algorithms using time and space complexity.

Course Code	**:**	**CIPC 301**
Course Title	**:**	**Design and Analysis of Algorithms**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	
Course Category	**:**	**PC**

Course Learning Objectives:

1. Able to design, implement and analysis of standard searching and sorting algorithms.
2. Implement standard divide and conquer, Dynamic programming, Greedy and backtracking algorithms.
3. Able to implement between different data structures i.e., trees, heaps etc. also, able to pick an appropriate data structure for any given design situation.
4. Able to implement the major graph algorithms and their analysis.

Course Content:

Unit-1 Introduction

Concept of Time and space complexity, analysis of algorithms, asymptotic notation, recurrence relations, design and analysis of D & C problems like quick sort etc, heap sort, priority queues, sorting in linear time, hashing, binary search trees.

Unit-2 Graph Algorithms

Graph representation & traversal (search), topological sort, strongly connected components, minimum spanning trees – Kruskal and Prim's, Single source shortest paths, relaxation, Dijkstra's algorithm, Bellman-Ford algorithm, single source shortest paths for directed acyclic graphs, all-pairs shortest path.

Unit-3 B-Trees and Dynamic programming

B-Trees: representation and operations; Elements of Dynamic Programming, structure and steps, Matrix-chain multiplication, longest common subsequence.

Unit-4 Greedy & Backtracking Approaches:

Greedy algorithms – Elements, activity-selection problem, Huffman codes, task scheduling problem, Knapsack Problem, Backtracking – Elements, 8 – Queens, Graph Coloring, Hamiltonian Cycles.

Text Books:

1. Cormen, Leiserson and Rivest: Introduction to Algorithms, 3/e, PHI.
2. Horowitz, Sahni, and Rajasekaran: Fundamentals of Computer Algorithms, Second Edition, Universities Press, Hyderabad.
3. Aho, Hopcroft, and Ullman: The Design and Analysis of Computer Algorithms, Addison Wesley.

Course Outcomes

1. Able to design, implement and analysis of standard searching and sorting algorithms.
2. Implement standard divide and conquer, Dynamic programming, Greedy and backtracking algorithms.
3. Able to implement between different data structures i.e., trees, heaps etc. also, able to pick an appropriate data structure for any given design situation.
4. Able to implement the major graph algorithms and their analysis.

Course Code	**:**	**CIPC 300**
Course Title	**:**	**Operating Systems**
Number of Credits and L/T/P scheme	**:**	**4 and 3/ 0/ 2**
Prerequisites (Course code)	**:**	
Course Category	**:**	**PC**

Course Learning Objectives

1. To understand the services and design of an operating system.
2. To understand the structure and organization of file system
3. To understand the process states and various concepts such as scheduling and synchronization related with it.
4. To understand different memory management approaches.
5. Students should be able to use system calls for managing processes, memory and file system.
6. students should understand the data structures and algorithms for implementation of OS.

Course Content

Unit-1: Computer system architecture and organization, Introduction and evolution of OS, Introduction to distributed OS, Real time systems and multimedia systems. OS structures: OS services, system calls and programs, OS design and implementation. Processes: Process concept, scheduling policies, algorithms, multilevel queuing, operations on process, Inter-process communication. Threads: multithreading models and threading issues. CPU scheduling: Criteria and algorithms, multiprocessor and thread scheduling.

Unit-2: Process synchronization: critical sections, classical two process and n-process solutions, hardware primitives for synchronization, semaphores, monitors, classical problems in synchronization (producer-consumer, readers-writer, dining philosophers, etc.).
Deadlocks: modelling, resource allocation, characterization, prevention and avoidance, detection and recovery.

Unit-3: Memory management: Swapping, contiguous memory allocation, paging, multilevel paging, segmentation, demand paging, page replacement algorithms, allocation of frames, thrashing, working set model. Input/Output: I/O system and services, device controllers and device drivers, disks, scheduling algorithms and management.

Unit-4: File system interface: access methods, access control, directory structures, file organization, file sharing and protection. system performance, protection and security, , OS design considerations for security, access control lists and OS support, internet and general network security. Operating system as service provider: Access control matrix, access control list, capability matrix, encryption and access permissions in Linux and Windows, Operating system based security threats: Viruses and worms, Trojan horses, trapdoors and keyloggers, Buffer overflow attacks, phishing and social engineering, Methods to prevent buffer overflows, logic bomb, Denial of service, virtual machines: history, benefits, building blocks, types of virtual machines and their implementation, virtualization and operating system components.

Reference Books:

1. A. Silberschatz, Peter B. Galvin and G. Gagne, "Operating System Concepts," (6th or newer edition), Wiley.
2. H. Brinch, "Operating System Principles," Prentice Hall of India.
3. Dhamdhere, "Systems programming & Operating systems," TataMcGraw Hil
4. A. N. Habermann, "Introduction to Operating System Design," Galgotia publication, New Delhi.
5. A.S. Tanenbaum, "Modern Operating Systems," Prentice Hall of India.

Course Outcomes:

At the end of the course student will be able to

1. Understand functions, structures and history of operating systems
2. Able to know the design issues associated with operating systems
3. Master various process management concepts such as scheduling, synchronization, multithreading and deadlocks
4. Understand the various concepts associated with memory management such as virtual memory, demand paging, page replacements algorithms
5. Be familiar with various protection and security mechanisms
6. Be familiar with virtualization and operating system components

Course Code	**:**	**CIPC 401**
Course Title	**:**	**Database Management Systems**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Set Theory**
Course Category	**:**	**PC**

Course Learning Objectives:

1. To understand data models and database systems.
2. To understand relational database theory,
3. To learn database design and its normalization.
4. To understand concepts of transaction, its processing, and management.
5. To learn concurrency control techniques.
6. To develop practical skill in using MySQL DBMS.

Course Content:

Unit-1 (Introduction)

Data, Database, Database management system, Historical background from file systems to Database Systems, Data Models, Relational Data Model, ER model, Schemas and Instances, Database users, DBMS architecture.

Unit-2 (Relational Model and Query Language)

Theory of Relational Database, Key Integrity constraint, Relational Algebra, Relational Calculus: Domain Relational Calculus, Tuple Relational Calculus, SQL, queries writing in SQL.

Unit-3 (Database Design)

Normalization, Database Anomalies, Functional Dependencies, Candidate and Super Key, Non-loss Decomposition, Dependency Preservation, First, Second, Third Normal, BCNF, etc.

Unit-4 (Transaction Processing)

Transaction concept, a simple transaction model, states, ACID Properties, implementation of ACID properties, Serializability.

Unit-5 (Concurrency Control)

Need for Concurrency, Lock-based protocols, Deadlock, Starvation, deadlock handling, time-stamp based protocols, validation-based protocols.

Text Books:

1. A Silberschatz, H.F. Korth & S. Sudarshan: Data Base System Concepts, Mc Graw Hill, 4^{th}, 5^{th} or 6^{th} edition.
2. Elmasri & Navathe : Fundamentals of Database Systems, 5^{th}, 6^{th}, or 7^{th} edition Pearson.

Course Outcomes

1. To Design and Implement a small database for applications using MySQL DBMS.
2. To apply the concepts of normalization to database design.

Course Code	**:**	**CIPE 402**
Course Title	**:**	**Advanced Machine Learning**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Machine Learning and Data Analytics (CSIC 221)**
Course Category	**:**	**PE**

Course Learning Objectives:

1. The major goal of the course is to understand recent advance concepts of machine learning and their applications in representative applications.
2. Able to design and implement various machine learning algorithms in real world applications.
3. Focuses on exploring advanced concepts and methodologies used in machine learning with practical applications.

Course Content:

Unit-1 **Introduction to Machine Learning:** Basic concepts in machine learning, Mathematics of machine learning, Overview of supervised and unsupervised learning, Some common discrete and continuous distributions, Multi-task techniques.

Unit -2 **Bayesian Modelling and Gaussian Processes:** Bayesian learning, Gaussian models, Bayesian statistics, Bayesian neural networks, Applications of machine learning in natural language processing.

Unit -3 **Basics of Deep Learning (DL):** Introduction of generative models, Deep neural networks, Regularization for Deep Learning, Recurrent Neural Networks, Backpropagation through time.

Unit -4 **Other Models and Networks:** Approximate inference, Variational autoencoders, Long short term memory, Attention networks, Memory networks, Optimisation for neural networks, Applications of deep networks.

Text Books:

1. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. MIT Press 2012.
2. Ethem Alpaydin, Introduction to Machine Learning, Second Edition, PHI, 2010.
3. Ian Goodfellow, Yoshua Bengio and Aaron Courville. Deep Learning. MIT Press 2016.

Reference Books:

1. Tom M. Mitchell, Machine Learning, McGraw-Hill International Edition, 1997.
2. Christopher M. Bishop F.R.Eng., Pattern Recognition and Machine Learning, Springer, 2006.
3. J. Grus., Data Science from Scratch, Second Edition, O'Reilly. 2019.
4. Douglas C. Montgomery, George C. Runger., Applied Statistics and Probability for Engineers, Third Edition, John Wiley & Sons, Inc., 2003.
5. T. Hastie, R. Tibshirani and J. Friedman., The Elements of Statistical Learning, Second Edition, Springer, 2009.

Course Outcomes

1. Explore the advance concepts of machine learning.
2. Understand advance machine learning algorithms and their applications in representative applications.
3. Develop an appreciation for what is involved in learning from data.
4. Skills to tackle complex machine learning problems and develop innovative solutions in various domains.

Course Code	**:**	**CIPE 404**
Course Title	**:**	**Artificial Intelligence**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Data Structures**
Course Category	**:**	**PE**

Course Learning Objectives

1. It aims to give an introduction to the fundamentals of artificial intelligence.
2. The course is designed to develop a basic understanding of problem solving, knowledge representation, reasoning and learning methods of AI.
3. To learn about major topics of artificial intelligence, their fundamental differences and applicability.

Course Content

Unit 1 **Introduction:** Introduction **to** AI, Foundations and history, problems, Nature and Structure of Intelligent Agents.

Unit 2 **Problem-Solving:** Solving Problems by Searching, Heuristic Search techniques, Metaheuristic Search, Beyond Classical Search, Adversarial Search, Constraint Satisfaction Problems.

Unit 3 **Knowledge, Reasoning, and Planning:** Logical Agents, First-Order Logic, Inference in First-Order Logic, Classical Planning, Knowledge Representation, Uncertain Knowledge and Reasoning.

Unit 4 **Learning and Agents:** Learning from Examples, Knowledge in Learning, Definition of agents, Agent architectures, multi-agent systems.

Text Books:

1. Russell, S. J. and Norvig, P., Artificial Intelligence: A Modern Approach, Pearson Education, 3rd Edition, 2014.
2. Elaine Rich, Kevin Knight and Shivashankar B Nair, Artificial Intelligence, Tata McGraw Hill Education Private Limited, 3rd Edition, 2009.

Reference Books:

1. Patterson, D. W., Introduction to Artificial Intelligence and Expert Systems, Prentice Hall of India, 2012.

Course Outcomes

1. Understand basic concepts of AI and students will be able to apply those concepts in the field of Artificial Intelligence.
2. Identify problems where artificial intelligence techniques are applicable.
3. Apply various AI search algorithms, knowledge representation, reasoning to real-world problems.
4. Design of systems that act intelligently and learn from experience.