Indira Gandhi University Meerpur, Rewari

(A State University established under Haryana Act No.29 of 2013)
Recognized u/s 12 (b) & 2(f) of UGC Act, 1956

SCHEME OF STUDIES AND EXAMINATION
Common for
B.TECH CSE (Artificial Intelligence and Machine Learning)

SEMESTER 3rd
Modified Scheme effective from 2024-25

COURSE CODE AND DEFINITIONS

Course Code	Definitions
L	Lecture
T	Tutorial
P	Practical
BSC	Basic Science Courses
ESC	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management courses
PCC	Professional core courses
OEC	Open Elective courses
LC	Laboratory course
MC	Mandatory courses
PROJ	Project

Scheme effective from 2024-25

B. TECH CSE (Artificial Intelligence and Machine Learning)

Scheme of Studies/Examination

Semester- 3

Sr. No.	Category	Course Code	Course Title	Hours per week			Total Contact Hrs. per week	Credit	Examination Schedule (Marks)				Duration of Exam (Hours)
				L	T	P			Internal Assessment	Theory	Practical	Total	
1	Professional Core Course	PCC-CSE-201 (common with B.Tech.(CSE) in semester 3)	Database Management Systems	3	0	0	3	3	25	75		100	3
2	Professional Core Course	PCC-CSE-203 (common with B.Tech.(CSE) in semester 3)	Data Structures & Algorithms	3	0	0	3	3	25	75		100	3
3	Professional Core Course	PCC-CSE-304 common with B.Tech.(CSE) in semester 6)	Artificial Intelligence	3	0	0	3	3	25	75		100	3
4	Professional Core Course	PCC-CSE-207 common with B.Tech.(CSE) in semester 3)	Python Programming	3	0	0	3	3	25	75		100	3
5	Basic Science Course	BSC-MATH-271	Mathematics for AI	3	0	0	3	3	25	75		100	3
6	Professional Core Course	PCC-CSE-208 common with B.Tech.(CSE) in semester 4)	Object Oriented Programming	3	0	0	3	3	25	75		100	3
7	Professional Core Course	LC-CSE-209 common with B.Tech.(CSE) in semester 3)	Database Management Systems LAB	0	0	2	2	1	25		25	50	3
8	Professional Core Course	LC-CSE-326 common with B.Tech.(CSE) in semester 6)	Artificial Intelligence Lab using Python	0	0	2	2	1	25		25	50	3
9	Professional Core Course	LC-CSE-213 common with B.Tech.(CSE) in semester 3)	Data Structures & Algorithms LAB Using C	0	0	2	2	1	25		25	50	3
10	Professional Core Course	LC-CSE-215 common with B.Tech.(CSE) in semester 3)	Python Programming LAB	0	0	2	2	1	25		25	50	3
11	Professional Core Course	LC-CSE-214 common with B.Tech.(CSE) in semester 4)	Object Oriented Programming LAB Using C++	0	0	2	2	1	25		25	50	3
							TOTAL	**23**	**275**	**450**	**125**	**850**	

Scheme effective from 2024-25

B. TECH CSE (Artificial Intelligence and Machine Learning)

Scheme of Studies/Examination

Semester- 4

Sr. No.	Category	Course Code	Course Title	Hours per week			Total Contact Hrs. per week	Credit	Examination Schedule (Marks)				Duration of Exam (Hours)
				L	T	P			Internal Assessment	Theory	Practical	Total	
1	Professional Core Course	PCC-CSE-202 common with B.Tech.(CSE) in semester 4)	Discrete Mathematics	3	0	0	3	3	25	75		100	3
2	Professional Core Course	PCC-CSE-204 common with B.Tech.(CSE) in semester 4)	Computer Organization & Architecture	3	0	0	3	3	25	75		100	3
3	Professional Core Course	PCC-CSE-206 common with B.Tech.(CSE) in semester 4)	Operating System	3	0	0	3	3	25	75		100	3
4	Professional Core Course	PCC-AIML-241	Introduction to Machine Learning	3	0	0	3	3	25	75		100	3
5	Professional Core Course	PCC-CSE-303 common with B.Tech.(CSE) in semester 5)	Computer Networks	3	0	0	3	3	25	75		100	3
6	Professional Core Course	PCC-CSE-307 common with B.Tech.(CSE) in semester 5)	Design & Analysis of Algorithms	3	0	0	3	3	25	75		100	3
7	Professional Core Course	LC-CSE-325 common with B.Tech.(CSE) in semester 5)	Design & Analysis of Algorithms Using C++	0	0	2	2	1	25	-	25	50	3
8	Professional Core Course	LC-CSE-212 common with B.Tech.(CSE) in semester 4)	Operating System LAB	0	0	2	2	1	25		25	50	3
9	Professional Core Course	LC-AIML-281	Introduction to Machine Learning Lab	0	0	2	2	1	25		25	50	3
10.	Professional Core Course	LC-CSE-323 common with B.Tech.(CSE) in semester 5)	Computer Networks Lab	0	0	2	2	1	25	-	25	50	3
11	Humanities & Social Sciences Courses	HS-150	Design Thinking Lab	0	0	2	2	1	25	-	25	50	
							TOTAL	**23**	**275**	**450**	**125**	**850**	

NOTE: At the end of 4th semester each student has to undergo Practical Training of 4/6 weeks in an Industry/ Institute/ Professional Organization/ Research Laboratory/ training Centre etc. and submit a typed report along with a certificate from the organization & its evaluation shall be carried out in the 5th Semester.

Database Management System

Course code	PCC-CSE-201				
Category	Professional Core Course				
Course Title	Database Management System				
Scheme and Credits	L	T	P	Credits	Semester 3(common with B.Tech (CSE) in semester 3)
	3	0		3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Objectives of the course

A. To understand the different issues involved in the design and implementation of a database system.
b. To study the physical and logical database designs, database modeling, relational, hierarchical, and network models
c. To understand and use data manipulation language to query, update, and manage a database
d. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
e. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit: 1

Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). **Data models:** Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Unit: 2

Relational query languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server.

Relational database design: Domain and data dependency, Armstrong's axioms, Normal forms,

Dependency preservation, Lossless design.

Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms

Unit: 3

Storage strategies: Indices, B-trees, hashing,

Transaction processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

Unit: 4

Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

Advanced topics: Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

Suggested books:

"Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

Suggested reference books

"Principles of Database and Knowledge – Base Systems", Vol 1 by J. D. Ullman, Computer Science Press.

"Fundamentals of Database Systems", 5th Edition by R. Elmasri and S. Navathe, Pearson Education

"Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley

Course Outcomes

1. For a given query write relational algebra expressions for that query and optimize the developed expressions
2. For a given specification of the requirement, design the databases using E R method and normalization.
3. For a given specification, construct the SQL queries for Open source and Commercial DBMS -MYSQL, ORACLE, and DB2.
4. For a given query optimize its execution using Query optimization algorithms
5. For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.
6. Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Data Structure & Algorithms

Course code	PCC-CSE-203				
Category	Professional Core Course				
Course Title	Data Structure & Algorithms				
Scheme and Credits	L	T	P	Credits	Semester 3(common with B.Tech(CSE) in semester 3)
	3	0		3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Objectives of the course:

1. To impart the basic concepts of data structures and algorithms.
2. To understand concepts about searching and sorting techniques
3. To understand basic concepts about stacks, queues, lists, trees and graphs.
4. To enable them to write algorithms for solving problems with the help of fundamental data structures

Unit 1:

Introduction: Basic Terminologies: Concept of Data Structure, Choice of right Data Structure, Algorithms , how to design and develop algorithm , Complexity of algorithm. Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, **Searching:** Linear Search and Binary Search Techniques and their complexity analysis.

Unit 2:

Stacks and Queues: Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation -corresponding algorithms and complexity analysis. queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis

Unit 3:

Linked Lists: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.
Trees: Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded BinaryTree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.
Unit 4:

Sorting and Hashing: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Selection Sort Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods.

Graph: Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis. Minimum Spanning Tree: Kruskal's Algorithm, Prim's Algorithm.

Suggested books:

"Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.

Suggested reference books:

Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company
"How to Solve it by Computer", 2nd Impression by R.G. Dromey, Pearson Education.

Course outcomes

1. For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness.
2. For a given Search problem (Linear Search and Binary Search) student will able to implement it.
3. For a given problem of Stacks, Queues and linked list student will able to implement it and analyze the same to determine the time and computation complexity.
4. Student will able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity.
5. Student will able to implement Graph search and traversal algorithms and determine the time and computation complexity.

ARTIFICIAL INTELLIGENCE

Course code	PCC-CSE-304				
Category	Professional Core Course				
Course title	Artificial Intelligence				
Scheme and Credits	L	T	P	Credits	Semester 3 (common with B.Tech(CSE) in semester 6)
	3	0	0	3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Objectives of the course:

- To provide historical perspective of AI and its foundation.
- To provide the most fundamental knowledge to the students so that they become familiar with basic principles of AI towards problem solving, inference, knowledge representation and learning.
- Explore application of AI techniques in Expert systems, Neural Networks.
- Explore the current trends, potential, limitations, and implications of AI.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1

Introduction: Definition of AI, History of AI, nature of AI problems, examples of AI problems. **Problem solving by search**: Uninformed Search: Depth First Search (DFS), Breadth First Search (BFS). Informed Search: Best First Search, A*. Local Search: Hill Climbing. Problem Reduction Search: AO*. Population Based Search: Ant Colony Optimization, Genetic Algorithm. Game Playing: MinMax Algorithm, Alpha-Beta Pruning.

UNIT 2

Knowledge Representation: Types of Knowledge, Knowledge Representation Techniques/schemes: Propositional Logic, Predicate Logic, Semantic nets, Frames. Knowledge representation issues. Rule based systems.

UNIT 3

Reasoning under Uncertainty: Basics of Probability Theory, Probabilistic Reasoning, Bayesian Reasoning, Dempster-Shafer Theory.

Planning: Introduction to Planning, Representation of Planning, Partial-order Planning.

Unit 4:

Learning: Introduction to Learning, Types of Learning: Learning by Induction, Rote Learning, Symbol Based Learning, Identification Trees, Explanation Based Learning, Transformational Analogy, Introduction to Neural Networks, Expert Systems, Current trends in Artificial Intelligence

Suggested Test books:

1. Artificial Intelligence: A Modern Approach Third Edition Stuart Russell and Peter Norvig, 2010, Pearson Education.

Suggested reference books:

1. Elaine Rich, Kevin Knight, & Shivashankar B Nair, Artificial Intelligence, McGraw Hill, 3rd ed.,2009.
2. Introduction to Artificial Intelligence & Expert Systems, Dan W Patterson, PHI.,2010.
3. Artificial intelligence, Patrick Henry Winston, 1992, Addition Wesley 3 Ed.

Course Outcomes:

1. Display the understanding of the historical perspective of AI and its foundation.
2. Apply basic principles of AI in solutions that require problem solving, inference, knowledge representation and learning.
3. Demonstrate fundamental understanding of various application of AI techniques in Expert systems, Neural Networks.
4. Demonstrate an ability to share in discussion of AI, it's the current trends, limitations, and implications of AI.

Python Programming

Course code	PCC-CSE-207				
Category	Professional Core Course				
Course Title	Python Programming				
Scheme and Credits	L	T	P	Credits	Semester 3(common with B.Tech(CSE) in semester 3)
	3	0	0	3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Objectives of the course:

- To impart the basic concepts of Python programming.
- To understand syntax of Python language
- To create dynamic applications in Python language.
- To implement object oriented concepts using Python language

Detailed contents:

Unit 1:

Introduction: Installing Python; basic syntax, interactive shell, editing, saving, and running a script; data types; variables, assignments; numerical types; arithmetic operators and expressions; Loops and selection statements, Control statements String manipulations: subscript operator, indexing, slicing a string; text files: reading/writing text and numbers from/to a file; creating and reading a formatted file

Unit 2:

Lists, dictionary and Design with functions: Basic list operators, replacing, inserting, removing an element; searching and sorting lists; dictionary literals, adding, and removing keys,

accessing and replacing values; traversing dictionaries. Hiding redundancy, complexity; arguments and return values; Program structure and design. Recursive functions.

Unit 3: **Simple graphics and image processing**: Simple graphics, Turtle operations, Manipulating turtle screen, Drawing two dimensional shapes, examining an object attributes, Taking a random walk, Color and RGB scheme, Image processing: Image manipulation operations, properties of images, image module, copying, blurring and reducing image. Graphical User Interfaces: Terminal based and GUI based programs, Simple GUI-Based Programs, Windows and Window Components, Input and Output with Entry Fields, Defining and Using Instance Variables, Other Useful GUI Resources.

Unit 4: Object Oriented concepts: Classes and OOP: classes, objects, attributes and methods; defining classes; design with classes, data modelling; persistent storage of objects, Inheritance, polymorphism, operator overloading; abstract classes; exception handling, try block. Multithreading: Threads and Processes, Sleeping Threads, Producer, Consumer, and Synchronization, The Readers and Writers Problem, Shared Cell Class, Thread-Safe Class

Course outcomes

- For a given conceptual problem student will able to analyze the problem and write a program in python with basic concepts.
- For a given problem of Strings and texts, student will able to analyze the problem and write a program in python with basic concepts involving strings and texts.
- The knowledge of list and dictionary will enable student to implement in python language and analyze the same.
- Student will able to write a program using functions to implement the basic concepts of object oriented programming language

Suggested books:

"Fundamentals of Python: First Programs" Kenneth Lambert, Course Technology, Cengage Learning, 2012

Suggested reference books:

"Introduction to Computer Science Using Python: A Computational Problem-Solving Focus",By Charles Dierbach, John Wiley & Sons, December 2012,

Mathematics for AI

Course code	BSC-MATH-271				
Category	Basic Science Course				
Course title	**Mathematics for AI**				
Scheme and Credits	L	T	P	Credits	Semester 3
	3	0	0	3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Objective of the course

1. To learn about functions of two and more variables, limit, continuity, partial derivatives and multiple integrals.
2. To learn finding maxima and minima of functions of two variables.
3. To learn the applications of double and triple integrals in determining areas and volumes.
4. To learn solving ordinary differential equations of first, second and higher order.
5. To understand and apply the fundamental theorems of vector calculus, including Green, Stokes, and Gauss theorems, to solve problems related to line, surface, and volume integrals.

Unit-1

Multivariable Differential Calculus: Functions of several variables: Limit and Continuity, Partial derivatives, Homogeneous functions, Euler's Theorem, Total derivative, Local maxima and minima for function of two variables, Lagrange's method of undetermined multipliers

Unit-2

Multivariable Integral Calculus: Double integral, Change of order of integration, Triple integral Change of variables (Cartesian to polar) for double and (Cartesian to Spherical and Cylindrical polar coordinates) for triple integrals, Applications: area by double integral and volume by triple integral

Unit-3

Differential Equations: First order linear equations, Bernoulli's equations, Exact differential

equations, Equations reducible to exact differential equations, Applications of differential equations of first order and first degree to simple electric circuits, Newton's law of cooling, Heat flow and Orthogonal trajectories**,** Linear differential equations of second and higher order, Complete solution,Complementary function and Particular integral, Method of variation of parameters to find particular integral, Cauchy's and Legendre's linear equations, Simultaneous linear differential equations with constant coefficients

Unit-4

Vector Calculus: Differentiation of vectors, Scalar and vector point functions, Gradient of a scalar field and Directional derivative, Divergence and Curl of a vector field, Integration of vectors, Line integral, Surface integral, Volume integral, Green, Stokes and Gauss theorems (without proof) and their applications.

Suggested Test books:

B. S. Grewal, Higher Engineering Mathematics, Khanna Publishers.

Suggested reference books:

1. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, Pearson Education.
2. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons.
3. B. V. Ramana, Higher Engineering Mathematics, Tata McGraw-Hill Publishing Company Limited.
4. N. P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications.
5. W. E. Boyce and R. C. Di Prima, Elementary Differential Equations and Boundary Value Problems, Wiley India.
6. S. L. Ross, Differential Equations, Wiley India.
7. E. A. Coddington, An Introduction to Ordinary Differential Equations, Prentice Hall India.

Course Outcomes

After completion of the course student will be able:

CO1. To define the concepts and terminology of multivariable calculus and differential equations including partial derivatives, multiple integrals, exact differential equations, linear and Bernoulli's equations, linear differential equations of second and higher order, gradient, divergence and curl.

CO2. To understand the significance and contribution of various concepts/ theorems/methods such as Euler's Theorem, Lagrange's method of undetermined multipliers, multiple integrals, change of order, change of variable, exactness of differential equations, method of variation of parameters, Green, Stokes and Gauss theorems etc.

CO3. To apply the notion of multivariable calculus in finding maxima and minima of functions, area and volume.

CO4. To solve ordinary differential equations of first, second and higher order.

Object Oriented Programming

Course code	PCC-CSE-208				
Category	Professional Core Course				
Course title	Object Oriented Programming				
Scheme and Credits	L	T	P	Credits	Semester-3(common with B.Tech(CSE) in semester 4)
	3	0	0	3	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Course Objectives

1. To understand how the C++ is superset of C by incorporating the Object oriented features in C language.
2. To apply the syntax and semantics of C++ Programming language.
3. To learn the designing of C++ classes.
4. To learn how to efficiently use the memory using Pointers and Dynamic Memory Management
5. To learn how to implement different types of constructors and the use of destructor.
6. To learn how to implement the concept of data abstraction and encapsulation.
7. To learn how to perform different types of overloading i.e. operators and functions.
8. To learn how inheritance helps to reuse the code.
9. To learn how we can implement dynamic binding with polymorphism.
10. To learn the use of exception handling in C++ programs.
11. To learn how to design and use of generic classes with templates.

Unit - I

Object-Oriented Programming Concepts: Introduction, comparison between procedural programming paradigm and object-oriented programming paradigm, basic concepts of object-oriented programming — concepts of an object and a class, interface and implementation of a class,

operations on objects, relationship among objects, abstraction, encapsulation, data hiding, inheritance, overloading, polymorphism, messaging.

Classes and Objects: Specifying a class, creating class objects, accessing class members, access specifiers, static members, use of const keyword, friends of a class, empty classes, nested classes, local classes, abstract classes, container classes, bit fields and classes.

Unit - II

Inheritance: Introduction, defining derived classes, forms of inheritance, ambiguity in multiple and multipath inheritance, virtual base class, object slicing, overriding member functions, object composition and delegation, order of execution of constructors and destructors.

Pointers and Dynamic Memory Management: Declaring and initializing pointers, accessing data through pointers, pointer arithmetic, memory allocation (static and dynamic), dynamic memory management using new and delete operators, pointer to an object, this pointer, pointer related problems - dangling/wild pointers, null pointer assignment, memory leak and allocation failures.

Unit - III

Constructors and Destructors: Need for constructors and destructors, copy constructor, dynamic constructors, explicit constructors, destructors, constructors and destructors with static members, initializer lists.

Operator Overloading and Type Conversion: Overloading operators, rules for overloading operators, overloading of various operators, type conversion - basic type to class type, class type to basic type, class type to another class type.

Virtual functions & Polymorphism: Concept of binding - early binding and late binding, virtual functions, pure virtual functions, abstract clasess, virtual destructors.

Unit - IV

Exception Handling: Review of traditional error handling, basics of exception handling, exception handling mechanism, throwing mechanism, catching mechanism, rethrowing an exception, specifying exceptions.

Templates and Generic Programming: Template concepts, Function templates, class templates, illustrative examples.

TEXT BOOKS, AND/OR REFERENCE MATERIAL:

1. BjraneStroustrup, “C++ Programming language”,3rd edition, Pearson education Asia(1997)
2. LaforeR.”Object oriented Programming in C++”,4th Ed. Techmedia,New Delhi(2002).
3. Yashwant Kenetkar,”Let us C++”,1stEd.,Oxford University Press(2006)
4. B.A. Forouzan and R.F. Gilberg,CompilerScience,”A structured approach using C++” Cengage Learning, New Delhi.

Course Outcomes

After the completion of the course, the students will

- Understand the concept of Object Oriented Programming through C++.
- Identify importance of object oriented programming and difference between Procedural programming and object oriented programming features.
- be able to make use of objects and classes for developing programs.
- be able to use various object oriented concepts to solve different problems.
- be able to develop the programs /Projects using some advanced features of C++ Programming.

Database Management System Lab

Course code	LC-CSE-209				
Category	Professional Core Course				
Course title	Database Management System Lab				
Scheme and Credits	L	T	P	Credits	Semester 3(common with B.Tech.(CSE) in semester 3)
	0	0	2	1	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

Course Objectives:

- Keep abreast of current developments to continue their own professional development
- To engage themselves in lifelong learning of Database management systems theories and technologies this enables them to purse higher studies.
- To interact professionally with colleagues or clients located abroad and the ability to overcome challenges that arises from geographic distance, cultural differences, and multiple languages in the context of computing.
- Develop team spirit, effective work habits, and professional attitude in written and oral forms, towards the development of database applications.

Contents:

i. Creation of a database and writing SQL queries to retrieve information from the database.
ii. Performing Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions.
iii. Creation of Views, Synonyms, Sequence, Indexes, Save point.
iv. Creating an Employee database to set various constraints.
v. Creating relationship between the databases.
vi. Study of PL/SQL block.
vii. Write a PL/SQL block to satisfy some conditions by accepting input from the user.
viii. Write a PL/SQL block that handles all types of exceptions.
ix. Creation of Procedures.
x. Creation of database triggers and functions

xi. Mini project (Application Development using Oracle/ MySQL)

a) Inventory Control System

b) Material Requirement Processing.

c) Hospital Management System.

d) Railway Reservation System.

e) Personal Information System.

f) Web Based User Identification System.

g) Time Table Management System.

h) Hotel Management

ARTIFICIAL INTELLIGENCE LAB USING PYTHON

Course code	LC-CSE-326				
Category	Professional Core Course				
Course title	Artificial Intelligence Lab Using Python				
Scheme and Credits	L	T	P	Credits	Semester 3(common with B.Tech(CSE) in semester 6)
	0	0	2	1	
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

List of Program:

1. Write a Program to Implement Breadth First Search using Python.
2. Write a Program to Implement Depth First Search using Python.
3. Write a Program to Implement Tic-Tac-Toe game using Python.
4. Write a Program to Implement 8-Puzzle problem using Python.
5. Write a Program to Implement Water-Jug problem using Python.
6. Write a Program to Implement Travelling Salesman Problem using Python.
7. Write a Program to Implement Tower of Hanoi using Python.
8. Write a Program to Implement Monkey Banana Problem using Python.
9. Write a Program to Implement Missionaries-Cannibals Problems using Python.
10. Write a Program to Implement 8-Queens Problem using Python.

Note: At least 5 to 10 more exercises to be given by the teacher concerned.

Data Structures and Algorithms Lab Using C

Course code	LC-CSE-213				
Category	Professional Core Course				
Course title	Data Structures and Algorithms Lab Using C				
Scheme and Credits	L	T	P	Credits	Semester-3(common with B.Tech(CSE) in semester 3)
	0	0	2	1	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

Data Structures Lab List of practical exercises, to be implemented in C Language.

1. Write a menu driven program that implements following operations (using separate functions) on a linear array:
    - Insert a new element at end as well as at a given position
    - Delete an element from a given whose value is given or whose position is given
    - To find the location of a given element
    - To display the elements of the linear array
2. Write a menu driven program that maintains a linear linked list whose elements are stored in on ascending order and implements the following operations (using separate functions):
    - Insert a new element
    - Delete an existing element
    - Search an element
    - Display all the elements
3. Write a program to demonstrate the use of stack (implemented using linear array) in converting arithmetic expression from infix notation to postfix notation.
4. Program to demonstrate the use of stack (implemented using linear linked lists) in evaluating arithmetic expression in postfix notation.
5. Program to demonstration the implementation of various operations on a linear queue represented using a linear array.
6. Program to demonstration the implementation of various operations on a circular queue represented using a linear array.
7. Program to demonstration the implementation of various operations on a queue

represented using a linear linked list (linked queue).

8. Program to illustrate the implementation of different operations on a binary search tree.
9. Program to illustrate the traversal of graph using breadth-first search
10. Program to illustrate the traversal of graph using depth-first search.
11. Program to sort an array of integers in ascending order using bubble sort.
12. Program to sort an array of integers in ascending order using selection sort.
13. Program to sort an array of integers in ascending order using insertion sort.
14. Program to sort an array of integers in ascending order using radix sort.
15. Program to sort an array of integers in ascending order using merge sort.
16. Program to sort an array of integers in ascending order using quick sort.
17. Program to sort an array of integers in ascending order using heap sort.
18. Program to sort an array of integers in ascending order using shell sort.
19. Program to demonstrate the use of linear search to search a given element in an array.
20. Program to demonstrate the use of binary search to search a given element in a sorted array in ascending order.

Python Programming Lab

Course code	LC-CSE-215				
Category	Professional Core Course				
Course title	Python Programming Lab				
Scheme and Credits	L	T	P	Credits	Semester-3(common with B.Tech.(CSE) in semester 3)
	0	0	2	1	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

Objectives

- To write, test, and debug simple Python programs.
- To implement Python programs with conditionals and loops.
- Use functions for structuring Python programs.
- Represent compound data using Python lists, tuples, and dictionaries.
- Read and write data from/to files in Python.

List of Programs

1. Compute the GCD of two numbers.
2. Find the square root of a number (Newton‘s method)
3. Exponentiation (power of a number)
4. Find the maximum of a list of numbers
5. Linear search and Binary search
6. Selection sort, Insertion sort
7. Merge sort
8. First n prime numbers
9. Multiply matrices
10. Programs that take command line arguments (word count)
11. Find the most frequent words in a text read from a file
12. Simulate elliptical orbits in Pygame
13. Simulate bouncing ball using Pygame

Outcome:

- Write, test, and debug simple Python programs.
- Implement Python programs with conditionals and loops

- Develop Python programs step-wise by defining functions and calling them.
- Use Python lists, tuples, dictionaries for representing compound data.
- Read and write data from/to files in Python.

Object Oriented Programming Lab Using C++

Course code	LC-CSE-214				
Category	Professional Core Course				
Course title	Object Oriented Programming Lab Using C++				
Scheme and Credits	L	T	P	Credits	Semester 3(common with B.Tech(CSE) in semester 4)
	0	0	2	1	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

Contents:

1. [Classes and Objects] Write a program that uses a class where the member functions are defined inside a class.
2. [Classes and Objects] Write a program that uses a class where the member functions are defined outside a class.
3. [Classes and Objects] Write a program to demonstrate the use of static data members.
4. [Classes and Objects] Write a program to demonstrate the use of const data members.
5. [Constructors and Destructors] Write a program to demonstrate the use of zero argument and parameterized constructors.
6. [Constructors and Destructors] Write a program to demonstrate the use of dynamic constructor.
7. [Constructors and Destructors] Write a program to demonstrate the use of explicit constructor.
8. [Initializer Lists] Write a program to demonstrate the use of initializer list.
9. [Operator Overloading] Write a program to demonstrate the overloading of increment and decrement operators.
10. [Operator Overloading] Write a program to demonstrate the overloading of binary arithmetic operators.
11. [Operator Overloading] Write a program to demonstrate the overloading of memory management operators.
12. [Inheritance] Write a program to demonstrate the multilevel inheritance.
13. [Inheritance] Write a program to demonstrate the multiple inheritance.
14. [Inheritance] Write a program to demonstrate the virtual derivation of a class.
15. [Polymorphism] Write a program to demonstrate the runtime polymorphism.
16. [Exception Handling] Write a program to demonstrate the exception handling.

17. [Templates and Generic Programming] Write a program to demonstrate the use of function template.

18. [Templates and Generic Programming] Write a program to demonstrate the use of class template.

Discrete Mathematics

Course code	PCC-CSE-202				
Category	Professional Core Course				
Course Title	Discrete Mathematics				
Scheme and Credits	L	T	P	Credits	Semester - 4(common with B.Tech.(CSE) in semester 4)
	3	0		3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit-I

Sets, Relation, Function and Propositional Logic: Operations and Laws of Sets, Cartesian Products, Representation of relations, Binary Relation, Equivalence Relation, Partial Ordering Relation, POSET, Hasse Diagram, Lattices and its types, Function, Bijective functions, Inverse and Composite Function, Finite and infinite Sets, Countable and Uncountable Sets, Cantor's diagonal argument and The Power Set theorem, **S**chroeder-Bernstein theorem, Propositions, Logical operations, Conditional Statements, Tautologies, Contradictions, Logical Equivalence, The use of Quantifiers

Unit-II

Basic Counting Techniques and Recurrence Relation: Pigeon-hole principle, Permutation and Combination**,** the Division algorithm: Prime Numbers, The GCD: Euclidean Algorithm, The Fundamental Theorem of Arithmetic., Linear recurrence relation with constant coefficients, Homogenous Solutions, Particular Solutions, Total Solutions, Solving recurrence relation using generating functions

Unit-III

Algebraic Structures: Definitions and examples of Algebraic Structures with one Binary Operation: Semi Groups, Monoids, Groups; Congruence Relation and Quotient Structures, Permutation Groups, Cyclic groups, Normal Subgroups, Definitions and examples of Algebraic Structures with two Binary Operation: Rings, Integral Domain, Fields; Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive
Normal Form

Unit-IV

Graphs and Trees: Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Multigraph and Weighted graph, Shortest path in Weighted graphs, Eulerian paths and circuits, Hamiltonian path and circuits, Planar Graphs, Euler's formulae, Graph Colouring, Trees, Binary trees and its traversals, Trees Sorting, Spanning tree, Minimal Spanning tree

Reference Books:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw – Hill
2. Satinder Bal Gupta: A Text Book of Discrete Mathematics and Structures, University Science Press, Delhi.
3. C. L. Liu and D. P. Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, Tata McGraw – Hill.
4. J.P. Tremblay and R. Manohar, Discrete mathematical structures with applications to computer science, TMG Edition, TataMcgraw-Hill
5. Discrete Mathematics, Babu Ram, Pearson Publication
6. Discrete Mathematics, Semyour Lipschutz and Marc Lipson, Schaum's outline

Course Outcomes

The students will learn

1. To solve mathematical problems based on concepts of set theory, relations, functions and lattices.
2. To express logic sentence in terms of quantifiers and logical connectives.
3. To apply basic counting techniques to solve permutation and combination problems.
4. To solve recurrence relations.
5. To classify algebraic structure of any given mathematical problem.
6. To evaluate Boolean functions and simplify expressions using the properties of Boolean algebra
7. To develop the given problem as graph networks and solve with techniques of graph theory.

Computer Organization & Architecture

Course code	PCC-CSE-204				
Category	Professional Core Course				
Course title	Computer Organization & Architecture				
Scheme and Credits	L	T	P	Credits	Semester-4(common with B.Tech(CSE) in semester 4)
	3	0	0	3	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Objectives of the course:

To expose the students to the following:

- How Computer Systems work & the basic principles
- Instruction Level Architecture and Instruction Execution
- The current state of art in memory system design
- How I/O devices are accessed and its principles.
- To provide the knowledge on Instruction Level Parallelism
- To impart the knowledge on micro programming
- Concepts of advanced pipelining techniques.

Unit 1

Data representation: Data Types, Complements, Fixed-Point Representation, Conversion of Fractions, Floating-Point Representation, Gray codes, Decimal codes, Alphanumeric codes, Error Detection Codes.

Register Transfer and Microoperations : Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Microoperations, Logic Microoperations, Shift Microoperations, Arithmetic Logic Shift Unit.

Unit 2

Basic Computer Organization and Design : Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instruction, Input-Output Instruction, Complete Computer Description, Design of Basic Computer, Design of Accumulator Logic.

Central Processing Unit : General Register Organization, Stack organization, Instruction Format, Addressing Modes, Data Transfer and Manipulation, Program Control, RISC, CISC.

Unit 3

Pipelining: Parallel Processing, Amdahl's law, Pipelining, Arithmetic Pipeline, Instruction Pipeline, Pipeline Hazards, RISC Pipeline.

Parallel Processors: Introduction to Parallel Processors, Concurrent access to memory andCache Coherency.

Vector Processing : Vector Operations, Memory Interleaving, Supercomputers, Array Processors: Attached Array Processor, SIMD Array Processor.

Unit 4

Input-output Organization : I/O deviceinterface, I/O transfers–program controlled, interrupt driven and DMA, Privileged and Non-Privileged Instructions, Software Interrupts.

Memory organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Associative Mapping, Direct Mapping, Set-Associative Mapping, Writing into Cache, Cache Initialization, Virtual Memory.

Suggested books:

1) "Computer System Architecture", 3rd Edition by M.Morris Mano, Pearson.
2) "Computer Organization and Design: The Hardware/Software Interface", 5th Edition by David A. Patterson and John L. Hennessy, Elsevier.
3) "Computer Organization and Embedded Systems", 6th Edition by CarlHamacher, McGraw Hill Higher Education.

Suggested reference books:

1) "Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB/McGraw-Hill
2) "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.
3) "Computer System Design and Architecture", 2nd Edition by Vincent P. Heuring and Harry F. Jordan, Pearson Education.

Course outcomes :

1) Draw the functional block diagram of a single bus architecture of a computer and describe the function of the instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set.
2) Write assembly language program for specified microprocessor for computing

   16 bit multiplication, division and I/O device interface (ADC, Control circuit, serial port communication).
3) Write a flowchart for Concurrent access to memory and cache coherency in Parallel Processors and describe the process.
4) Given a CPU organization and instruction, design a memory module and analyze its operation by interfacing with the CPU.
5) Given a CPU organization, assess its performance, and apply design techniques to enhance performance using pipelining, parallelism and RISC methodology.

Operating System

Course code	PCC-CSE-206				
Category	Professional Core Course				
Course title	Principles of Operating System				
Scheme and Credits	L	T	P	Credits	Semester-4(common with B.Tech(CSE) in semester 4)
	3	0	0	3	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1:

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services.

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. Thread: Definition, Various states, Benefits of threads, Types of threads, Multithreading.

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS, SJF, SRTF, RR Scheduling.

UNIT 2:

Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, The Producer\ Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc.

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, and Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

UNIT 3:

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Optimal Page Replacement and Least Recently used (LRU).

UNIT 4:

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks. Case study on UNIX and WINDOWS Operating System.

Comparative Study of Latest Operating System: Evolution, Architecture and Characteristics of various Operating systems like MS-Windows, Ubuntu, Mac OS, Fedora, Solaris, Free BSD, Chrome OS, CentOS, Debian, Deepin

Suggested books:

- Operating System Concepts Essentials, 9th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
- Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

Suggested reference books:

- Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
- Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley
- Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India
- Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

Course Outcomes:

CO1: Understand the structure and architectural components of OS to analyze and design the applications to run in parallel. Moreover, students would be able to develop scheduling algorithms to optimize various parameters like CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time for research purpose.

CO2: Understand the design issues associated with Operating system (e.g. Mutual exclusion, Deadlock detection etc.) to gain insight towards developing algorithms/techniques for efficient deadlock handling.

CO3: For a given specification of memory organization, develop the techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.

CO4: Design and implement file management system for a given specification. Identify, use and evaluate the disk management policies with respect to various performance evaluation parameters.

Introduction to Machine Learning

Course code	PCC-AIML-241				
Category	Professional Core Course				
Course title	Introduction to machine learning				
Scheme and Credits	L	T	P	Credits	Semester 4
	0	0	2	1	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Course Outcomes (CO):

By the end of the Course, Students will be able to:

CO1: Define various fundamental concepts, type and other related terms of machine leaning. **Comparative study** of various machine learning models (Level 1, 2: Remember and Understand)

CO2: Apply the machine learning models to various real world problems (Level 3: Apply)

CO3: Create various machine learning models and techniques with the help of existing machine learning algorithm for various real world problems. (Level 6: Create)

CO4: Analyze and evaluate Performance measurement of models in terms of different parameters like accuracy, Confusion matrix, ROC, AUC, MAD etc (Level 4, 5: Analyze and Evaluate)

Course Content

UNIT 1:

Machine Learning: Definition, History, Need, Features, Block diagrammatic representation of learning machines, Classification of Machine Learning: Machine Learning life cycle, Applications of Machine Learning.

UNIT 2:

Supervised learning, Unsupervised learning, Reinforcement Learning, Supervised Learning: Definition, how it works. Types of Supervised learning algorithms: k-Nearest Neighbours, Naïve Bayes, Linear Regression, Logistic Regression.

UNIT 3:

Unsupervised Learning: Definition, how it works. Difference between supervised and unsupervided

learing, Types of unsupervised learning algorithms: Classification and Clustering, K-means.

UNIT 4:

Performance measurement of models in terms of accuracy, confusion matrix, precision & recall, F1-score, receiver Operating Characteristic Curve (ROC) curve and AUC, Median absolute deviation (MAD), Distribution of errors.

Suggested books

1. E. Alpaydin, Introduction to Machine Learning, Prentice Hall of India, 2006.
2. T Hastie, R Tibshirani and J Friedman, The Elements of Statistical Learning Data Mining, Inference, and Prediction, 2nd Edition, Springer, 2009.
3. C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2010.

Suggested reference books

1. R. O. Duda, P. E. Hart, and D.G. Stork, Pattern Classification, John Wiley and Sons, 2012.
2. Simon O. Haykin, Neural Networks and Learning Machines, Pearson Education, 2016

COMPUTER NETWORKS

Course code	PCC-CSE-303				
Category	Professional Core Course				
Course title	Computer Networks				
Scheme and Credits	L	T	P	Credits	Semester 4(common with B.Tech(CSE) in semester 5)
	3	0	0	3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Course Objectives:

- To develop an understanding of modern network architectures from a design and
- Performance perspective.
- To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
- To provide an opportunity to do Network programming
- To provide a WLAN measurement idea.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit: 1

Introduction: Data communication, Components, Computer networks and its historical development, distributed processing, Internet

Network Models: OSI model and TCP/IP Model

Physical Layer – physical layer functions, Data Representation, Simplex, Half Duplex and Full Duplex Transmission, Modulation and Multiplexing, Packet and circuit switching, Transmission media, Topologies, connectionless and connection-oriented services.

Data Link Layer : Data link layer functions and services, MAC Addressing, Framing, Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window Protocol.

Unit: 2

Medium Access Control: MAC layer functions, Random access, Controlled Access and channelization protocols.

Network Layer: Network layer functions and services, Logical addressing, IPv4 classful and classless addressing, subnetting, NAT, IPv4, ICMPv4, ARP, RARP and BOOTP, IPv6, IPv6 addressing, DHCP.

Network Devices: Repeater, hub, switch, router and gateway.

Unit: 3

Routing Algorithms: introduction to routing, Shortest Path Algorithm, Flooding, Hierarchical Routing, Link State and Distance Vector Routing

Transport Layer: Transport layer functions and services, Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), TCP connection management

Application Layer: Application layer functions and services, Domain Name Space (DNS), EMAIL, File Transfer Protocol (FTP), HTTP, SNMP

Unit: 4

Congestion Control, Quality of Service, QoS Improving techniques.
LAN: Ethernet, Token Bus, Token Ring, MAN Architecture- DQDB, WAN Architectures- Frame Relay, ATM, SONET/SDH

Network Security: Firewalls, security goals, types of attack, Introduction to cryptography, Types of ciphers: symmetric and asymmetric key ciphers.

Suggested books:

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw-Hill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.

Suggested reference books:

1. Computer Networks, 8th Edition, Andrew S. Tanenbaum, Pearson New International Edition.
2. Internetworking with TCP/IP, Volume 1, 6th Edition Douglas Comer, Prentice Hall of India.
3. TCP/IP Illustrated, Volume 1, W. Richard Stevens, Addison-Wesley, United States of America.

Course Outcomes:

- Explain the functions of the different layer of the OSI Protocol.
- Draw the functional block diagram of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) and describe the function of each.
- Identify and connect various connecting components of a computer network.
- Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open source available software and tools.

DESIGN AND ANALYSIS OF ALGORITHMS

Course code	PCC-CSE-307				
Category	Professional Core Course				
Course title	Design and Analysis of Algorithms				
Scheme and Credits	L	T	P	Credits	Semester 4(common with B.Tech(CSE) in semester 5)
	3	0		3	
Internal Assessment	25 Marks				
Exam	75 Marks				
Total	100 Marks				
Duration of Exam	03 Hours				

Course Objectives:

- Analyze the asymptotic performance of algorithms.
- Write rigorous correctness proofs for algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.
- Synthesize efficient algorithms in common engineering design situations.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have questions from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit 1:

Introduction to Algorithms: Algorithm, Performance Analysis (Time and Space complexity), Asymptotic Notation (Big OH, Omega and Theta)-best, average and worst-case behaviour. Elementary Data Structures (Basic terminology of Stacks and Queues, Tree, Graph), Sets and Disjoint Set Union.

Divide and Conquer: General method, Binary Search, Merge Sort, Quick Sort, and other sorting algorithms with divide and conquer strategy, Strassen's Matrix Multiplication algorithms and analysis of these problems.

Unit 2:

Greedy Method: General method, Fractional Knapsack problem, Job Sequencing with Deadlines, Minimum Cost Spanning Trees, Single source shortest paths.

Dynamic Programming: General method, Optimal Binary Search Trees, 0/1 knapsack, The Traveling Salesperson problem.

Unit 3:

Back Tracking: General method, The 8-Queen's problem, Sum of subsets, Graph Colouring,

Hamiltonian Cycles.

Branch and Bound: The method, 0/1 knapsack problem, Traveling Salesperson problem, Efficiency considerations.

Unit 4:

NP Hard and NP Complete Problems: Basic concepts, Cook's theorem, NP hard graph problems, NP hard scheduling problems, NP hard code generation problems, and Some simplified NP hard problems.

Suggested Text Books:

1. Fundamental of Computer algorithms, Ellis Horowitz and Sartaj Sahni, 1978, Galgotia Publication
2. Introduction to Algorithms, Thomas H Cormen, Charles E Leiserson and Ronald L Rivest: 1990, TMH

Suggested Reference Books:

1. The Design and Analysis of Computer Algorithm, Aho A.V. Hopcroft J.E., 1974, Addison Wesley.
2. Algorithms-The Construction, Proof and Analysis of Programs, Berlion, P.Bizard, P., 1986. Johan Wiley & Sons,
3. Writing Efficient Programs, Bentley, J.L., PHI
4. Introduction to Design and Analysis of Algorithm, Goodman, S.E. &Hedetnieni, 1997, MGH.
5. Introduction to Computers Science- An algorithms approach, Jean Paul Trembley, Richard B.Bunt, 2002, T.M.H.
6. Fundamentals of Algorithms: The Art of Computer Programming Vol Knuth, D.E.: 1985, Naresh Publication.

Course Outcomes:

- To identify and justify correctness of algorithms and to analyse running time of algorithms based on asymptotic analysis.
- To understand when an algorithmic design situation calls for the divide-and-conquer paradigm. Synthesize divide-and-conquer algorithms.
- Describe the greedy paradigm and dynamic-programming paradigm. Explain when an algorithmic design situation calls for it.
- Developing greedy algorithms/dynamic programming algorithms, and analyze it to determine its computational complexity.
- To write the algorithm using Backtracking and Branch and Bound strategy to solve the problems for any given model engineering problem.

DESIGN & ANALYSIS OF ALGORITHMS USING C++

<table>
<tr><td>Course code</td><td colspan="5">LC-CSE-325</td></tr>
<tr><td>Category</td><td colspan="5">Professional Core Course</td></tr>
<tr><td>Course title</td><td colspan="5">Design & Analysis of Algorithms Using C++</td></tr>
<tr><td rowspan="2">Scheme and Credits</td><td>L</td><td>T</td><td>P</td><td>Credits</td><td rowspan="2">Semester 4(common with B.Tech(CSE) in semester 5)</td></tr>
<tr><td>0</td><td>0</td><td>2</td><td>1</td></tr>
<tr><td>Internal Assessment</td><td colspan="5">25 Marks</td></tr>
<tr><td>Exam</td><td colspan="5">25 Marks</td></tr>
<tr><td>Total</td><td colspan="5">50 Marks</td></tr>
<tr><td>Duration of Exam</td><td colspan="5">03 Hours</td></tr>
</table>

Course Objectives:

- Implementation of various algorithms and to analyze the performance of algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.
- Synthesize efficient algorithms in common engineering design situations.

List of programs:

1. Write a Program for iterative and recursive Binary Search.
2. Write a Program to sort a given set of elements using the Quick Sort/Merge Sort/Selection Sort method and determine the time required to sort the elements.
3. Write a Program for implementation of Fractional Knapsack problem using Greedy Method and 0/1 Knapsack problem using Dynamic Programming.
4. Write a Program to find the shortest path from a given vertex to other vertices in a weighted connected graph using Dijkstra's algorithm.
5. Write a Program to find the minimum cost spanning tree (MST) of a given undirected graph using Kruskal's algorithm/Prim's Algorithms.
6. Write a Program to implement N-Queens problem using back tracking.
7. Write a Program to check whether a given graph is connected or not using DFS method.
8. Write a program to implement the Travelling Salesman Problem (TSP).

Note: At least 5 to 10 more exercises to be given by the teacher concerned.

Course Outcomes:

- The course will help in improving the programming skills of the students.
- The design of algorithms for any problem will inculcate structured thinking process in the students and improve the analytical power.

Operating System Lab

Course code	LC-CSE-212				
Category	Professional Core Course				
Course title	Operating System Lab				
Scheme and Credits	L	T	P	Credits	Semester 4(common with B.Tech(CSE) in semester 4)
	0	0	2	1	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

Contents:

1 Introduction to UNIX File System.

2. File and Directory Related Commands in UNIX.

3. Essential UNIX Commands for working in UNIX environment.

4. I/O Redirection and Piping

5. Introduction to VI Editors.

6. Introduction of Processes in UNIX

7. Communication in UNIX and AWK.

8. Introduction of the concept of Shell Scripting.

9. Decision and Iterative Statements in Shell Scripting.

10. Writing the Shall Scripts for unknown problems.

Suggested Books:

1. UNIX Shell Programming by YashavantKanetkar.
2. UNIX Concepts and Applications by Sumitabha Das

Course Outcomes.

Co1: Understand the structure and architectural components of UNIX Operating System to analyze and design the problem. Moreover, students would be able to know the Basic Introduction of UNIX Operating System.

Co2: Basic Introduction of UNIX Commands that are used for operating the UNIX.

Co3: Introduction of Shell Scripting and VI Editor.so that the students get familiar with writing the UNIX scripts in UNIX editor.

Co4: Students will establish themselves as effective professionals by solving real problems with UNIX Shell Scripting knowledge and with attention to teamwork, critical thinking and problem solving skills by Writing Shell Scrips of unknown problems

Introduction to Machine Learning Lab

Course code	LC-AIML-281				
Category	Professional Core Course				
Course title	Introduction to machine learning Lab				
Scheme and Credits	L	T	P	Credits	Semester 4
	0	0	2	1	
Branches (B. Tech.)	Artificial Intelligence and Machine Learning				
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

Contents:

1. Introduction to data repository sites like kaggle, UCI etc.
2. Introduction to python.
3. Write some basic script in python
4. Write the scripts plotting of data
5. Study of different library used for machine learning.
6. Implementation of simple regression model for a real life problem
7. Implementation of logistic regression model for a real life problem
8. Develop a model for k-means algorithm
9. Develop a model for k-nearest algorithm
10. Study and plotting of confusion matrix
11. Study and plotting performance matrices to measure the performance of a model

Course Outcomes:

By the end of the Course, Students will be able to:

CO1: Analyse the trends in datasets using descriptive statistics (Level 4: Analyse)

CO2: Implement machine learning algorithms using modern machine learning tools. (Level 3: Apply)

CO3: Evaluate machine learning algorithms for a given problem on the basis of different parameters. (Level 5: Evaluate)

CO4: Create lab record for assignments that includes problem definitions, design of solutions and conclusions and **Demonstrate** use of ethical practices, self-learning, and team spirit. (Level 3: Apply, Level 6: Create)

Suggested books

- Machine learning using Python by U Dinesh Kumar, Publisher : Wiley
- Machine learning with Python by Abhisek vijayvergia, Publisher: BPB

COMPUTER NETWORKS LAB

Course code	LC-CSE-323				
Category	Professional Core Course				
Course title	Computer Networks Lab				
Scheme and Credits	L	T	P	Credits	Semester 4 (common with B.Tech(CSE) in semester 5)
	0	0	2	1	
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

Hands-on experiments related to the course contents of PCC-CSE-303 using hardware resources and using simulation tool.

Design Thinking Lab

Course code	HS-150				
Category	Humanities and Social Sciences Course				
Course title	Design Thinking Lab				
Scheme and Credits	L	T	P	Credits	Semester 4
	0	0	2	1	
Internal Assessment	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

COURSE OBJECTIVE(S):

The objective of this Course is to provide the new ways of creative thinking and Learn the innovation cycle of Design Thinking process for developing innovative products which useful for a student in preparing for an engineering career.

COURSE CONTENTS:

Unit 1: An Insight to Learning

Understanding the Learning Process, Kolb's Learning Styles, Assessing and Interpreting

Unit 2: Remembering Memory

Understanding the Memory process, Problems in retention, Memory enhancement techniques

Unit 3: Emotions: Experience & Expression

Understanding Emotions: Experience & Expression, Assessing Empathy, Application with Peers

Unit 4: Basics of Design Thinking

Definition of Design Thinking, Need for Design Thinking, Objective of Design Thinking, Concepts & Brainstorming, Stages of Design Thinking Process (explain with examples) –Empathize, Define, Ideate, Prototype, Test

Unit 5: Being Ingenious & Fixing Problem

Understanding Creative thinking process, Understanding Problem Solving, Testing Creative Problem Solving

Unit 6: Process of Product Design

Process of Engineering Product Design, Design Thinking Approach, Stages of Product Design, Examples of best product designs and functions, Assignment – Engineering Product Design

Unit 7: Prototyping & Testing

What is Prototype? Why Prototype? Rapid Prototype Development process, Testing, Sample Example, Test Group Marketing

Unit 8: Celebrating the Difference

Understanding Individual differences & Uniqueness, Group Discussion and Activities to encourage the understanding, acceptance and appreciation of Individual differences

Unit 9: Design Thinking & Customer Centricity

Practical Examples of Customer Challenges, Use of Design Thinking to Enhance Customer Experience, Parameters of Product experience, Alignment of Customer Expectations with Product Design

Unit 10: Feedback, Re-Design & Re-Create

Feedback loop, Focus on User Experience, Address "ergonomic challenges, User focused design,

rapid prototyping & testing, final product, Final Presentation – “Solving Practical Engineering Problem through Innovative Product Design & Creative Solution”.

Course Outcomes (CO):

Student will able to:

1. Compare and classify the various learning styles and memory techniques and Apply them in their engineering education
2. Analyze emotional experience and Inspect emotional expressions to better understand users while designing innovative products
3. Develop new ways of creative thinking and Learn the innovation cycle of Design Thinking process for developing innovative products
4. Propose real-time innovative engineering product designs and Choose appropriate frameworks, strategies, techniques during prototype development
5. Perceive individual differences and its impact on everyday decisions and further Create a better customer experience