Syllabus

Post Graduate Diploma in Computer Application (PGDCA)

2022-23

"Pacific Hills", Pratap Nagar Extension, Airport Road, Debari, Udaipur (Raj.) | Ph. : 0294-2494520 Fax : 0294-2492641
Email : fca@pacific-university.ac.in | www.pacific-university.ac.in

Program Outcomes for PGDCA

PO 1Students are eligible to pursue MCA (Lateral Entry) and apply for jobs in various multinational companies, industries, banks.

PO 2:They can start their own business in web development and software development.

PO 3Students are able to use their knowledge to develop different web and windows based applications.

PO 4: Students can create database, websites and applications for their clients.

PO 5: Students can also pursue the career of computer operators.

PO 6: Students can also become network administrators.

Program Specific Outcomes (PSO):

1) Students become eligible to pursue MCA and M.Sc. in Information Technology

2) They can also join MBA.

3) Ability to apply knowledge of layered network Models, their protocols and technologies in building network and Internet based applications.

4) Design, test, develop and maintain desktop, web, mobile and cross platform software applications using modern tools and technologies.

5) Prepare graduates who will thrive to pursue life-long learning to fulfill their goals.

6) Prepare graduates who will be successful professionals in industry, government, academic research, entrepreneurial pursuit and consulting firms.

Program Education Outcome (PEO)

PSO1: The students acquire knowledge about basics and fundamentals of information technology, basic programming concepts of procedure oriented and object oriented languages (C and Java), fundamentals of web programming (HTML, CSS, JavaScript and PHP), Database management system, computer networking and computer based accounting information.

PSO2:Students learn to develop and debug codes in different languages.

PSO3Students are able to design web based applications using PHP, HTML, DHTML, CSS and JavaScript.

PSO4: In order to enhance programming skills of the young IT professionals, the concept of project development in using the technologies learnt during the semester has been introduced.

PSO5: To enhance knowledge in robotics, provide experimental hardware equipment for teaching the basics of robotics, robot dynamics and control, and robot system design and application.

PSO6: To enhance logical ability and programming concepts by implementing programming lab.

PSO7: Preparing students for future aspects by building and improving their creativity, social awareness, and general knowledge.

PSO8: Encouraging students to convert their start-up idea to reality by implementing.

PSO9: Ability to understand the changes or future trends in the field of computer application.

PSO10: Ability to identify, formulate, analyse and solve problems of programming using different languages.

PACIFIC UNIVERSITY

FACULTY OF COMPUTER Science

POST GRADUATE DIPLOMA IN COMPUTER APPLICATIONS
SEMESTER-I

S. No.	Course Code	Subject Name	Period			Evaluation Scheme		
			L	T	P	ISE	ESE	Total
1	PGDCA -101	Computer Fundamental and Architecture	3	1	0	30	70	100
2	PGDCA -102	Programming in C	3	1	0	30	70	100
3	PGDCA -103	Database Management System	3	1	0	30	70	100
4	PGDCA -104	Accounting & Financial Management	3	1	0	30	70	100
5	PGDCA -105	Discrete Mathematics	3	1	0	30	70	100
6	PGDCA -151	Office Management Lab (DOS & M S office)	0	0	4	30	70	100
7	PGDCA -152	DBMS Lab	0	0	4	30	70	100
8	PGDCA -153	Programming in C Lab	0	0	4	30	70	100
9	PGDCA -154	Seminar	0	0	4	30	70	100
			15	5	16	Total		900

Course Code	**PGDCA-101**
Course Title	**Computer Fundamental and Architecture**
Credits	4
Course Objective(s)	A **Computer Fundamentals** course equips students with a basic understanding of hardware and software, essential computing skills like file management and internet use, and an introduction to programming and networking. It emphasizes cyber security and ethical practices, preparing learners for advanced studies or practical applications in technology.
Course Outcome (CO)	**Co1**: **Understand Basic Computer Concepts:** Introduce the history and evolution of computers. Explain the fundamental principles of computing and digital systems. **CO2**: **Familiarize with Computer Hardware:** Identify and describe the components of a computer system (e.g., CPU, memory, storage, input/output devices).Understand the functions and interactions of hardware components. **CO3**: **Learn Software Fundamentals:** Differentiate between system software and application software. Understand operating systems and their roles in managing hardware and software resources **CO4**: **Develop Basic Computing Skills:** Perform word processing, spreadsheet manipulation, and presentations using productivity tools. Understand internet navigation, email usage, and basic online safety practices. **CO5**: **Understand Computer Networks:** Learn basic networking concepts, including LAN, WAN, and the Internet. Understand how data is transferred between systems. **CO6: Introduce Programming Basics:** Learn fundamental programming concepts and logic. Explore introductory-level coding using a simple programming language.

PGDCA-101 Computer Fundamental and Architecture

Unit-1

Computer Basics: Algorithms, Flowchart, A Simple Model of a Computer, Characteristics of Computers, Problem-solving Using Computers. **Data Representation**: Representation of Integers, Character, Fractions and Real number. Code: BCD, EBCDIC, Gray, ASCII-7 &ASCII-8.

Number System: Binary Addition, Subtraction (1's and 2's Compliment), Multiplication and Division . Conversion: Decimal to Binary, Decimal to Octal, Decimal to Hexadecimal, Binary to Decimal, Binary to Octal, Binary to hexadecimal, Octal to Decimal, Octal to binary, hexadecimal to Binary and Decimal .

Unit-2

Computer Memory: Memory Cell, Memory Organization, Read Only Memory, Serial Access Memory, Physical Devices Used to Construct Memories, Magnetic Hard Disk, floppy Disk Drives, CD-ROM, Magnetic Tape Drives, Blu-Ray Disk, USB, DVD. **Computer Architecture**: Interconnection of Units, Processor to Memory communication, I/O to Processor Communication, Interrupt Structures, Multiprogramming, Processor Features, Reduced Instruction Set Computers(RISC),Introduction to Virtual Memory.

Unit-3

Software and Communication Concepts: Types of Software, System Software, Application Software, Utility Software, Language Translators, Booting and diagnostic test, Programming Languages. Communication: Mode of Communication, Basic Data Communication Model, Type of Data transmission, Transmission Media-Guided and unguided, Modem, Computer Networks and Topology.

Unit-4

Combinational Digital Circuits: Gates, Boolean Functions and Expressions, Designing Gate Networks, Combinational Circuits (Adder, Subtractor, Multiplexer, Demultiplexer, Encoder, Decoder) and Sequential Circuits, Useful Sequential Parts,
Programmable Sequential Parts, Flip-Flops (R-S, J-K, and T Flip Flop), Introduction to Registers .

Unit-5

Computer System Technology: Block Diagram of Computer, Computer Systems and their Parts, Generations, Peripherals I/O and Communications, Software Applications.

References:

1. William Stalling, "Computer Organization & Architecture", Pearson education Asia
2. Mano Morris, "Computer System Architecture", PHI
3. Zaky&Hamacher, "Computer Organization", McGraw Hill
4. B. Ram, "Computer Fundamental Architecture & Organization", New Age.
5. Tannenbaum, " Structured Computer Organization", PHI.
6. M S Dulawat& Sanjay Gaur "Introduction to Information Technology", Himanshu Publications New Delhi

Course Code	**PGDCA-102**
Course Title	**Programming in C**
Credits	4
Course Objective(s)	A **Computer Fundamentals** course equips students with a basic understanding of hardware and software, essential computing skills like file management and internet use, and an introduction to programming and networking. It emphasizes cyber security and ethical practices, preparing learners for advanced studies or practical applications in technology.
Course Outcome (CO)	**CO1: Understand the Fundamentals of C Programming:** Learn the basic structure, syntax, and semantics of the C language. **CO2:Develop Problem-Solving Skills:** Use algorithms and flowcharts to solve problems and translate them into C programs **CO3: Master Key Programming Concepts:** Understand and apply concepts like variables, data types, operators, control statements, loops, and functions. **CO4: Learn Memory Management:** Gain knowledge of pointers, dynamic memory allocation, and their applications in programming. **CO5:Understand Modular Programming:** Explore the use of functions, recursion, and modular design for efficient code development. **CO6: Work with Data Structures in C:** Learn about arrays, strings, structures, and file handling for data organization and manipulation. **CO7: Develop Debugging and Testing Skills:** Use debugging techniques and tools to identify and fix errors in C programs.

PGDCA-102 Programming in C

UNIT 1

Problem Solving with Computers: Algorithms, and Flowcharts. Data types, constants, variables, operators

UNIT 2

Data input and output, assignment statements, conditional statements, string and character handling.

UNIT 3

Iteration, arrays, strings processing, defining function, types of functions, function prototype, passing parameters, recursion. Storage class specifies, pre-processor, header files and standard functions.

UNIT 4

Pointers: Definition and uses of pointers, pointer arithmetic, Call by Value, Call by reference, pointers and array, pointer to pointer. Structures, union, Introduction to Dynamic Memory Allocation.

UNIT 5

Data files: Opening, closing, creating, processing and unformatted data files,command line arguments.

References:

1. Gottfried, “Programming in C”, Schaum’s Series, Tata McGraw Hill
2. Kernighan, Ritchie, “The C Programming Language”, PHI
3. YashwantKanitkar, “Working with C”, BPB
4. YashwantKanitkar, “Pointer in C”, BPB
5. YashwantKanitkar, “Let us C”, BPB
6. Bajpai, Kushwaha, Yadav, “Computers & C Programming”, New Age
7. E. Balagurusamy, “Programming in ANSI C”, TMH

Course Code	**PGDCA-103**
Course Title	**Database Management System**
Credits	4
Course Objective(s)	The course outcomes for a Database Management Systems (DBMS) program aim to provide students with a solid understanding of database concepts, design, and implementation. Students will learn to create and manage relational databases using SQL, apply normalization techniques for data integrity, and understand transaction management and security measures..
Course Outcome (CO)	**CO1**: **Understand Database Concepts:**Learn the fundamental concepts of databases, including data models, schemas, and architectures. **CO2**: **Explore Data Modeling:** Understand and design entity-relationship (ER) diagrams to represent real-world scenarios. **CO3**: **Learn Database Design Principles:**Master normalization techniques to create efficient and well-structured databases. **CO4**: **Familiarize with SQL:** Gain proficiency in Structured Query Language (SQL) for data definition, manipulation, and retrieval. **CO5**: **Understand Transaction Management:** Learn the principles of transaction processing, concurrency control, and recovery techniques. **CO6**: **Implement Relational Databases:** Develop and manage relational databases using modern DBMS tools. **CO7**: **Learn Database Administration:**Gain insight into database security, backup, recovery, and user management.

PGDCA-103 Database Management System

UNIT-I

Overview of DBMS: Basic DBMS terminology, data base system v/s file system, data independence, Data abstraction, Architecture of a DBMS

UNIT-II

Introduction to data models: entity relationship model, hierarchical model: from network to hierarchical, relational model, comparison of network, hierarchical and relational models.

UNIT-III

Data modeling using the Entity Relationship Model: ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation, reduction of an ER diagrams to tables, extended ER model, relationships of higher degree.

UNIT-IV

Normalization: normal forms, first, second, third normal forms, BCNF

Transaction Processing Concepts: Transaction system, Recovery from transaction failures, log based recovery, checkpoints, deadlock handling.

Concurrency Control Techniques: Concurrency control, locking Techniques for concurrency control, Time stamping protocols for
Concurrency control, validation based protocol,
Replication and allocation techniques for distributed system, overview
of concurrency control and recovery in distributed database.

UNIT-V

Introduction to SQL: Characteristics of SQL, Advantages of SQL, SQL data types and literals, Types of SQL commands, SQL operators
And their procedure, Tables, views and indexes, Queries and sub queries, Aggregate functions, insert, and update and delete operations, Joins,
Unions, Intersection, Minus in SQL.

References

1 Date C J, “An Introduction To Database System”, Addision Wesley
2 Korth, Silbertz, Sudarshan, “Database Concepts”, McGraw Hill
3 Elmasri, Navathe, “Fundamentals Of Database Systems”, Addision Wesley
4 Paul Beynon Davies, “Database Systems”, Palgrave Macmillan
5 Bipin C. Desai, “An introduction to Database Systems”, Galgotia Publication
6 Majumdar& Bhattacharya, “Database Management System”, TMH
7 Ramakrishnan, Gehrke, “Database Management System”, McGraw Hill
8 Bharti P.K, “An introduction to Database Systems”, JPNP

Course Code	**PGDCA-104**
Course Title	**Accounting & Financial Management**
Credits	4
Course Objective(s)	Upon completing the Accounting and Financial Management course, students will understand key accounting principles, analyze financial statements, and manage budgets effectively. They will be equipped to evaluate investments, apply ethical considerations, and use accounting software proficiently. Additionally, students will develop strong communication skills to present financial information clearly, preparing them for successful careers in the field.
Course Outcome (CO)	**CO1**: **Understanding Fundamental Principles**: To provide students with a solid foundation in the principles and concepts of accounting and financial management, including the accounting cycle, financial statements, and the role of accounting in business decision-making. **CO2**: **Financial Statement Analysis**: To equip students with the skills to analyze and interpret financial statements, enabling them to assess an organization's financial health and performance. **CO3**: **Budgeting and Forecasting**: To teach students how to prepare budgets and financial forecasts, helping them understand the importance of planning and control in financial management. **CO4**: **Cost Management**: To introduce students to cost accounting techniques and methods for managing and controlling costs within an organization. **CO5**: **InvestmentAnalysis**: To provide knowledge on evaluating investment opportunities, including understanding risk and return, and applying various valuation techniques.: To provide knowledge on evaluating investment opportunities, including understanding risk and return, and applying various valuation techniques. **CO6** :**Capital Structure and Financing Decisions**: To explore the concepts of capital structure, sources of finance, and the impact of financial decisions on an organization's value. **CO7**: **Use of Financial Software**: To familiarize students with accounting and financial management software tools that enhance efficiency and accuracy in financial reporting and analysis.

PGDCA-104 Accounting & Financial Management

UNIT – I

Definition of Accounting and its advantages & limitations, Scope of accounting, Branches of Accounting – Financial Accounting

UNIT – II:

Types of Accounts and Rules for Debit and Credit. Cash and Credit Transaction, Cash discount and Trade discount. Preparation ofJournal, ledger.

UNIT – III

Business and Management: Business Meaning and Contents, Business as a system,

Business Environment. Management Concept and Nature, Management Process, Basic function of Management, Management Level, Role of Manager.

UNIT – IV

Organizational Behavior: Need of Understanding human behavior in organization,

Challenges and Opportunities for OB. Management by Objective (MBO

UNIT- V

Managing Personnel:

Motivation Theories & Practices, Leadership Concept theories & Style, Compensation

Management.

Suggested books

1. B.P. Singh & T.N. Chabbra, “Business Organization and Management Functions”

, DhanpatRai& Co. 2000.

2. P.C Tripathi& P. N. Reddy, “Principles of Management”, Tata McGraw Hill

Publishing Company New Delhi.

3. L.M. Prasad, “Principles and Practices of Management”.

Course Code	**PGDCA-105**
Course Title	**Discrete Mathematics**
Credits	4
Course Objective(s)	Upon completing the Discrete Mathematics course, students will have a solid understanding of key concepts essential for computer science and mathematics. They will demonstrate proficiency in logical reasoning, apply set theory and combinatorial techniques to solve problems, and analyze graphs and algorithms effectively. Students will also tackle recursion and recurrence relations, model real-world challenges using discrete structures, and enhance their critical thinking and problem-solving skills. Ultimately, they will be equipped to communicate mathematical ideas clearly, preparing them for advanced studies and careers in their fields. successful careers in the field.
Course Outcome (CO)	**CO1: Understanding of Mathematical Logic**: Students will learn the principles of logic, including propositional and predicate logic, and how to construct and evaluate logical arguments. **CO2:-Set Theory Proficiency**: Students will gain a solid understanding of set theory, including operations on sets, relations, and functions, and their applications in various mathematical contexts. **CO3:Combinatorial Techniques**: Students will develop skills in counting techniques, including permutations, combinations, and the principles of inclusion-exclusion, to solve combinatorial problems. **CO4: Graph Theory Knowledge**: Students will explore the concepts of graph theory, including types of graphs, graph traversal algorithms, and applications in computer science.: Students will explore the concepts of graph theory, including types of graphs, graph traversal algorithms, and applications in computer science. **CO5: Understanding of Algorithms**: Students will learn about algorithmic thinking and the basics of algorithm design and analysis, including time complexity and efficiency.

PGDCA-105 Discrete Mathematics

Unit-I

Introduction to Discrete Mathematical Structures, : Sets & Relations- Sets, Types of Sets, Multi Sets, Operations on Sets, Relations and Properties of Relations, Representation of Relations, Equivalence Relation,

Unit-II

Combinatorics: Permutations and Combinations, Pigeon Hole Principle, Principle of Inclusion and Exclusion.

Unit-III

Mathematical Logic, Posets and Lattices: Partial, Lattices, Principle of Duality, Bounded, Distributed, and Complemented Lattices, Logic and Proposition Calculus-Basic Logical propositions, Propositions and truth table,Tautologies,contradictions,

Unit-IV

Graphs Theory: Basic Introduction of Graphs- Types of Graphs, Path and Circuits, Eulerian Path and Circuits, Hamiltonian Path and Circuits, Shortest Path Algorithms,

Unit-V

Finite State Machines and Languages: Grammar and Languages- Phrase structure Grammar, Types of Grammars and Languages, Finite
State Machines and Languages,

References

1. Liptschutz, Seymour, “Discrete Mathematics”, TMH
2. Trembley, J.P & R. Manohar, “Discrete Mathematical Structure with Application to Computer Science”, TMH
3. Kenneth H. Rosen, ” Discrete Mathematics and its applications”, TMH
4. Doerr Alan &Levasseur Kenneth, “Applied Discrete Structures for Computer Science”, Galgotia Pub. Pvt.Ltd
5. Gersting,“Mathematical Structure for Computer Science”,WH Freeman & Macmillan
8. C.L.Liu, “Elements of Discrete Mathematics”, McGraw Hill”
9. Peter Grossman, “Discrete Mathematics for Computer”, Palgrave Macmillan

Course Code	**PGDCA-151**
Course Title	**Office Management Lab (DOS & M S office)**
Credits	2
Course Objective(s)	Upon completing the Office Management Lab course, students will gain essential skills in office administration, including proficiency in office software, effective communication, and knowledge of standard office procedures. They will enhance teamwork and problem-solving abilities through collaborative projects, develop time management techniques for task prioritization, and embody professionalism and ethics, preparing them for success in administrative roles.
Course Outcome (CO)	**CO1:Practical Application of Office Skills**: Equip students with hands-on experience in essential office management skills, including document preparation, data entry, and office organization. **CO2:**Familiarity with Office Software: Provide proficiency in using various office software applications (e.g., word processing, spreadsheets, presentation software) to enhance productivity and efficiency in office tasks. **CO3: Communication Skills Development**: Enhance written and verbal communication skills through the preparation of business correspondence, reports, and presentations. **CO4:Understanding Office Procedures**: Familiarize students with standard office procedures, including filing systems, record management, and scheduling, to ensure smooth office operations. CO5: **Teamwork and Collaboration**: Foster teamwork and collaboration skills by engaging students in group projects and activities that simulate real office environments. CO6: **Problem-Solving and Decision-Making**: Develop critical thinking and problem-solving abilities by presenting students with real-world office scenarios that require effective decision-making.

List of Lab Programs:

1. What are the key features of Microsoft Excel that can enhance data management in an office setting?
2. How can effective business communication improve team collaboration and productivity?
3. What are the best practices for organizing and maintaining a filing system in an office?
4. What techniques can be used to prioritize tasks effectively in a busy office environment?
5. Describe a scenario where teamwork is essential in completing an office project. What roles might team members play?
6. How would you handle a conflict between team members in an office setting?
7. What are the key elements of professionalism in the workplace, and why are they important?
8. Discuss the importance of ethical behavior in office management. How can unethical behavior impact an organization?
9. How has technology changed the landscape of office management in recent years?
10. What tools or software can be used to manage office projects effectively?

Course Code	**PGDCA-152**
Course Title	DBMS Lab
Credits	2
Course Objective(s)	The DBMS lab course equips students with practical skills in database management, including SQL query optimization, database design, and transaction management. By completing a real-world project, they will enhance their problem-solving abilities and become familiar with various database management tools, preparing them for careers in the field.
Course Outcome (CO)	**CO1:Hands-On SQL Proficiency**: To provide students with practical experience in writing and executing SQL queries for data manipulation, retrieval, and management in a relational database environment. **CO2:Database Design Implementation**: To enable students to design and implement relational databases based on theoretical concepts learned in lectures, including creating tables, defining relationships, and applying normalization techniques. **CO3:DataModelling**: To practice creating Entity-Relationship (ER) diagrams and translating them into physical database schemas. **CO4:Database Administration Tasks**: To familiarize students with basic database administration tasks, such as user management, backup and recovery procedures, and performance monitoring.

List of Lab Programs:

1. Write a SQL query to retrieve all records from a given table.
2. How do you filter records using the WHERE clause?
3. Write a SQL query to sort results in ascending and descending order.
4. How can you use the DISTINCT keyword in a query?
5. Explain the difference between INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL OUTER JOIN.
6. Write a SQL query that uses a JOIN to combine data from two tables.
7. How do you perform a self-join on a table?
8. Write a SQL query to calculate the total, average, minimum, and maximum values of a column.
9. How do you use the GROUP BY clause in SQL?
10. Write a SQL query that groups data and filters groups using the HAVING clause.

Course Code	**PGDCA-153**
Course Title	**Programming in C Lab**
Credits	2
Course Objective(s)	Upon completing the Programming in C lab, students will be proficient in writing and debugging C programs, designing algorithms, and utilizing basic data structures. They will also gain skills in memory management, file handling, and modular programming, preparing them to effectively solve programming challenges.
Course Outcome (CO)	**CO1:Fundamental Concepts**: To understand the fundamental concepts of programming, including data types, variables, operators, control structures, and functions in the C programming language. **CO2:Problem-Solving Skills**: To develop problem-solving skills by designing algorithms and implementing them in C to solve various computational problems. **CO3:Hands-On Experience**: To provide hands-on experience in writing, compiling, and debugging C programs using an integrated development environment (IDE) or command-line tools. **CO4:Data Structures**: To introduce basic data structures such as arrays, strings, structures, and pointers, and to understand their applications in programming. **CO5: Memory Management**: To learn about memory management in C, including dynamic memory allocation and deallocation using functions like **malloc**, **calloc**, and **free**. **CO6: File Handling**: To understand file handling in C, including reading from and writing to files, and managing file operations. **CO7:Debugging Techniques**: To develop debugging skills and learn how to use debugging tools to identify and fix errors in C programs. **CO8:Project Development**: To work on small projects that integrate various concepts learned throughout the course, fostering teamwork and collaboration.

LIST of Lab Program:-

1. Write a C program to find the largest element in an array.
2. Write a C program to check if a given number is a prime number.
3. Write a C program to implement a simple calculator (addition, subtraction, multiplication, division).
4. Write a C program to sort an array using the bubble sort algorithm.
5. Write a C program to count the number of vowels in a given string.
6. Write a C program to find the GCD of two numbers using recursion.
7. Write a C program to merge two arrays into a third array.
8. Write a C program to find the Fibonacci series up to a given number.

Course Code	**PGDCA-108**
Course Title	**Seminar**
Credits	2
Course Objective(s)	Upon completion of the seminar course, participants will have gained a thorough understanding of key concepts and contemporary issues related to the topic. They will demonstrate improved critical thinking, research, and communication skills, and will be able to apply theoretical knowledge to real-world scenarios. Additionally, participants will enhance their ability to collaborate and network with peers, preparing them for future academic and professional pursuits.
Course Outcome (CO)	**CO1: Knowledge Acquisition:** To equip participants with a deep understanding of the seminar topic, including foundational theories, key concepts, and current trends. **CO2:Critical Thinking:** To develop participants' ability to analyze, evaluate, and synthesize information from various sources, fostering independent thought and critical reasoning. **CO3:Research Skills:** To enhance participants' research capabilities by engaging them in literature reviews, data analysis, and the exploration of relevant case studies. **CO4: Effective Communication:** To improve participants' oral and written communication skills through presentations, discussions, and the preparation of seminar papers or reports**.** **CO5: Collaboration:** To promote teamwork and collaborative learning by involving participants in group projects, discussions, and peer evaluations

PACIFIC UNIVERSITY

FACULTY OF COMPUTER Science

POST GRADUATE DIPLOMA IN COMPUTER APPLICATIONS

S.No.	Course Code	Subject Name	Period			Evaluation Scheme		
			L	T	P	ISE	ESE	Total
1	PGDCA-201	Web Technology & Development	3	1	0	30	70	100
2	PGDCA-202	Object oriented programming with C++& JAVA	3	1	0	30	70	100
3	PGDCA-203	Operating System	3	1	0	30	70	100
4	PGDCA-204	Data Communication and Computer Networks	3	1	0	30	70	100
5	PGDCA-205	System Analysis and Design	3	1	0	30	70	100
6	PGDCA-251	Web Technology& Development Lab	0	0	4	30	70	100
7	PGDCA-252	C++ Lab	0	0	4	30	70	100
8	PGDCA-253	Network Lab	0	0	4	30	70	100
9	PGDCA-254	Minor Project	0	0	4	200	100	300
			15	5	16			1100

Course Code	**PGDCA-201**
Course Title	**Web Technology & Development**
Credits	4
Course Objective(s)	The "Web Technology & Development" course teaches students to create dynamic websites using HTML, CSS, and JavaScript. They will learn server-side programming, database integration, and web security, while also using Git for version control. By the end, students will compile a portfolio of projects, preparing them for careers in web development.
Course Outcome (CO)	**CO1**: **Understand Web Fundamentals**: Grasp the core concepts of web technologies, including HTML, CSS, and JavaScript, and their roles in web development. **CO2**: **Develop Responsive Web Pages**: Create visually appealing and user-friendly web pages that adapt to various devices and screen sizes using responsive design techniques. **CO3**: **Implement Client-Side Scripting**: Utilize JavaScript and relevant libraries/frameworks (such as jQuery or React) to enhance user interaction and improve the overall user experience. **CO4**: **Work with Backend Technologies**: Gain an understanding of server-side programming languages (such as Node.js, Python, or PHP) and frameworks, enabling students to build and manage server-side applications. **CO5**: **Database Integration**: Learn to integrate databases (such as MySQL, MongoDB, or Firebase) into web applications for data storage, retrieval, and management. **CO6**: **Utilize Version Control Systems**: Familiarize students with version control systems (such as Git) to manage code changes and collaborate effectively in team environments. **CO7**: **Deploy Web Applications**: Understand the deployment process of web applications, including hosting options, domain management, and continuous integration/continuous deployment (CI/CD) practices.

PGDCA-201 Web Technology & Development

Unit-1

The internet: World Wide Web, web browser, Web Server.
Introduction of HTML: introduction, mark-up language, editing HTML, common tags, headers, text styles, linking, images, formatting text, horizontal rules and more line breaks, unordered lists, nested and ordered lists, basic HTML tables.

Unit-2

Formatting: basic HTML forms, HTML frames, internal linking, image maps.
Dynamic HTML: CSS : introduction – inline styles, creating style sheets with the style element, conflicting styles, linking external style sheets, backgrounds, element dimensions.

Unit-3

Dynamic HTML: object model and collections: introduction, Dynamic style, Dynamic Positioning.

Dynamic HTML: event model: introduction, event ON CLICK, event ON LOAD

Unit-4

Java script – introduction to scripting, Advantages of java script: introduction-java script in head & body section, memory concepts- Operators- decision making. Java script control structures.
Java script functions: introduction – program modules in java script - function definitions, duration of identifiers, scope rules, java script global functions.

Unit-5

Java script arrays: introduction, array-declaring and allocating arrays
Java script objects: introduction, math, string, data, Boolean and number objects.
Introduction to PHP – Advantages of PHP – Functions – Data types

References:

1. ShishirGundavarma, “CGI Programming on the World Wide Web”, O’Reilly & Associate.
2. DON Box, “Essential COM”, Addison Wesley.
3. Burdman, “Collaborative Web Development”, Addison Wesley.
4. Sharma &Sharma, “Developing E-Commerce Sites”, Addison Wesley
5. Ivan Bayross, “Web Technologies Part II”, BPB Publications.

Course Code	**PGDCA-202**
Course Title	**Object oriented programming with C++& JAVA**
Type of Course	Major
L T P	3:1:0
Credits	4
Course Objective(s)	Upon completing the Object-Oriented Programming with C++ course, students will grasp essential OOP concepts like classes, inheritance, and polymorphism. They will be able to design and implement classes, manage memory effectively, and handle errors. Additionally, students will complete a practical project that demonstrates their ability to apply these concepts, preparing them for careers in software development and further studies in programming.
Course Outcome (CO)	**CO1**: **Understand OOP Principles**: Grasp the fundamental concepts of OOP, including classes, objects, inheritance, polymorphism, and encapsulation. **CO2**: **Design and Implement Classes**: Create and manage classes and objects to model real-world entities effectively. **CO3**: **Utilize Inheritance and Polymorphism**: Apply inheritance to promote code reuse and polymorphism to enhance flexibility in code design. **CO4**: **Memory Management**: Understand dynamic memory allocation and deallocation, including the use of constructors and destructors. **CO5**: **Implement Error Handling**: Use exception handling to manage errors and ensure robust program execution. **CO6**: **Develop Software Projects**: Complete a project that demonstrates the application of OOP principles in a practical context.(CI/CD) practices.

PGDCA-202 Programming through C++

UNIT I

Overview of OOP: Definition and need for OOP, Differences between OOP and Procedure-Oriented Programming, Key principles of OOP: Abstraction, Classes and Objects, Encapsulation, Inheritance and Data Binding, Polymorphism, C++ Basics, Structure of a C++ program, Data types ,Declaration of variables, Expressions and Operators, Type Conversions ,Introduction to C++ Streams ,Flow Control Statements

Functions: Classification of Functions, Parameter Passing, Recursive Functions, Default Arguments, Inline Functions, Arrays and Strings, Structures

UNIT II

C++ Classes and Data Abstraction: Class Definition, Class structure, Class objects, Class scope this pointer, Static class members ,Constant member functions ,Constructors and Destructors ,Dynamic creation and destruction of objects ,Friend functions and classes ,Static class members, Overloading ,Function Overloading. Operator Overloading**,** Unary Operators, Binary Operators.

UNIT III

Inheritance: Defining a Class Hierarchy, Different forms of inheritance, Defining base and derived classes, Access to base class members, Base and derived class construction Destructors, Virtual base class.

UNIT IV:

Polymorphism: Static and Dynamic Bindings, Base and derived class virtual functions ,Dynamic binding through virtual functions ,Virtual function call mechanism ,Pure virtual functions, Abstract classes ,Exception Handling, Benefits of exception handling ,Throwing an exception, The try block, Catching an exception.

UNIT V

Advanced Topics: Templates, Function Templates, Class Templates, File Handling, Stream classes hierarchy, Stream I/O, File streams, Opening and closing data files, creating a data file, Reading and writing files, Error handling during file operations

Recommended Books:

1. BjarneStroustrup, “C++ Programming Language”, Addison Wesley
2. Balagurusamy E, “Object Oriented Programming with C++”, TMH, 2001
3. Booch Grady, “Object Oriented Analysis and Design with application 3/e”, Pearson
4. Lipman, Stanley B, JonsceLajole, “ C++ Primer Reading”, AWL, 1999
5. Horowitz and Sahani, “Fundamentals of data Structures”, Galgotia
6. R. Kruse etal, “Data Structures and Program Design in C” Pearson Education
7. A M Tenenbaumetal, “Data Structures using C & C++”, PHI

References

1. BjarneStroustrup, “C++ Programming Language”, Addison Wesley
2. Balagurusamy E, “Object Oriented Programming with C++”, TMH, 2001
3. Booch Grady, “Object Oriented Analysis and Design with application 3/e”, Pearson
4. Lipman, Stanley B, JonsceLajole, “ C++ Primer Reading”, AWL, 1999
5. Horowitz and Sahani, “Fundamentals of data Structures”, Galgotia
6. R. Kruse etal, “Data Structures and Program Design in C” Pearson Education
7. A M Tenenbaumetal, “Data Structures using C & C++”, PHI
8. Lipschutz, “Data Structure”, TMH
9. K Loudon, “Mastering Algorithms With C”, Shroff Publisher & Distributors
10. Bruno R Preiss, “Data Structures and Algorithms with Object Oriented Design Pattern in C++”, JhonWiley
11. Adam Drozdek, “Data Structures and Algorithms in C++”, Thomson Asia
12. Pal G. Sorenson, “An Introduction to Data Structures with Application”, TMH.

Course Code	**PGDCA-203**
Course Title	**Operating System**
Credits	4

Course Objective(s)	Upon completing the Operating Systems course, students will understand key concepts like process and memory management, file systems, and security. They will gain practical skills in implementing operating system functionalities, analysing performance, and addressing concurrency issues, preparing them to contribute effectively to system development and management.
Course Outcome (CO)	**CO1**: **Understanding Core Concepts:** To provide students with a solid foundation in the fundamental concepts of operating systems, including processes, threads, scheduling, memory management, and file systems. **CO2**: **Process Management:** To enable students to understand and implement process management techniques, including process creation, synchronization, and inter-process communication. **CO3**: **Memory Management:** To teach students about memory allocation strategies, virtual memory, paging, and segmentation, and how these concepts impact system performance. **CO4**: **File Systems:** To familiarize students with file system organization, management, and access methods, as well as the principles of data storage and retrieval. **CO5**: **Concurrency and Synchronization:** To equip students with the knowledge of concurrency issues and synchronization mechanisms, such as semaphores, mutexes, and monitors. **CO6**: **System Security and Protection:** To introduce students to the principles of operating system security, including user authentication, access control, and protection mechanisms.

PGDCA-203 Operating System

Unit I

Operating system: Introduction, Operating System Structure, Operating System Components and services, Goals of Operating System, Types of Operating System, Batch Operating System, Multiprogramming, Multiprocessing, Multitasking Operating System, Timesharing Operating System, Real Time Operating System, Distributed Operating System, Parallel System, and System calls,

Unit II

Processor management: Process overview, process states, Process Control Block, Process Scheduling, Scheduling Objectives, Scheduling Criteria of CPU, CPU scheduling algorithms; first come first serve, shortest Job First, Priority and Round Robin Scheduling, Multithreading Models

Unit III

Memory management: Address Binding, Logical versus Physical Address Space, contiguous Memory Allocation, fragmentation. Partition, paging and segmentation. Process loading and swapping, Page Replacement Algorithm

File Management: File Attributes, File Operation, File Types access methods, Directory systems-single level, tree structured, File Sharing, Disk Scheduling FCFS, SSTF, SCAN, C-SCAN.LOOK

Unit IV

Protection and Security: Protection, Goals of Protection, Domain of Protection, Domain Structure, Access matrix, language based Protection, File Protection, The security Problem, Authentication, Program Threads, Trojan Horse and Trap door System Threads Worms and Viruses, and Encryption

Unit V

Deadlock: Deadlock Characterization, Resource allocation graph, Deadlock Prevention, Deadlock Avoidance, safe state, Banker's Algorithm, safety algorithm, Deadlock Detection. Critical section Problem, Peterson's Solution, Semaphore.

Case Study: Window NT-Design principles, system Components, File System, Networking and programs, Linux History, Design Principles and components

References

1. Abraham Siberschatz and Peter Baer Galvin, "Operating System Concepts", Fifth Edition, Addision-Wesley
2. Milan Milankovic, "Operating Systems, Concepts and Design", McGraw-Hill.
3. Harvey M Deital, "Operating Systems", Addison Wesley

Course Code	**PGDCA-204**
Course Title	**Data Communication and Computer Networks**
Credits	4
Course Objective(s)	Upon completing the Data Communications and Computer Networks course, students will understand key networking principles and technologies. They will be proficient in analysing and designing network architectures, configuring and troubleshooting various networks, and implementing essential protocols. Additionally, students will be equipped to address security challenges and apply best practices for secure communications, preparing them for roles in network design and management. Development and management.
Course Outcome (CO)	**CO1**: **Fundamental Concepts**: Understand the basic principles of data communication and networking, including key terminologies, protocols, and architectures. **CO2**: **Network Models**: Explore the OSI and TCP/IP models, examining the functions of each layer and how they interact to facilitate communication. **CO3**: **Transmission Media**: Analyze different types of transmission media (wired and wireless) and their impact on network performance and reliability. **CO4**: **Networking Protocols**: Gain knowledge of essential networking protocols (e.g., TCP, UDP, IP) and their roles in data transmission and error handling. **CO5**: **Network Design and Implementation**: Develop skills in designing, configuring, and troubleshooting various types of networks, including LANs, WANs, and the Internet. **CO6**: **Current Trends**: Stay informed about emerging technologies and trends in data communications and networking, such as IoT, cloud computing, and software-defined networking.

"Pacific Hills", Pratap Nagar Extension, Airport Road, Debari, Udaipur (Raj.) | Ph. : 0294-2494520 Fax : 0294-2492641
Email : fca@pacific-university.ac.in | www.pacific-university.ac.in

Unit I

Data Communication : Basics of Data Communications, Components of Communication, Communication Model, Data Representation, Communication Modes, Network Basics & criteria, uses of computer networks, physical structure, network topology, Basics of LAN, WAN, MAN, Internet, Protocols and Standards.

Unit II

OSI Model T &CP/IP Model and Protocols: Layered Tasks, Layered Architecture, Design Issues for the Layer, OSI Model- Physical Layer, Data Link Layer, Network Layer, Transport Layer, Session Layer, Presentation Layer, Application Layer. TCP/IP Model, TCP/IP Protocols, Addressing- Logical Addressing, Physical Addressing. Connection Oriented and Connection less Services.

Unit III

Physical Layer and Media & Data Link Layer: Data and Signals, Transmission Impairment and Performance, Multiplexing and Spreading, Transmission Media, Switching, Introduction to PSTN. Introduction to Error, Types of Error, Error Detection and Correction, Hamming Distance, Framing, Flow Control, Error Control. Multiple Accesses- Random Access, Controlled Access, Channelization.

Unit IV

Network Layer & Transport Layer: IPv4 Addresses, IPv6 Addresses, Comparison b/w IPv4 and IPv6, Routing, Introduction to Routing Protocols. Flow Control, Buffering, UDP, TCP. Standards- IEEE standard, Ethernet, IEEE 802.11, Bluetooth, Wireless LANs. Connecting Devices, Virtual LAN, Introduction to SONET.

Unit V

Application Layer & Network Management and Security : DNS, Remote Logging, E-Mail, File Transfer, WWW, HTTP. Network Management System, SNMP. Cryptography, Security Services, IPSec, Firewalls, VPN, Email Security, Web Security.
SSL and VPN, Introduction only to firewalls and Kerberos, Cyber Laws.

References

1. A. S Tanenbaum, “Computer Networks, 3rd Edition”, PHI
2. W. Stallings, “Data and Computer Communication”, Macmillan Press
3. Comer, “Computer Networks & Internet”, PHI.
4. Comer, “Internetworking with TCP/IP”, PHI
5. Forouzan, “Data Communication and Networking”, TMH

Course Code	**PGDCA-205**
Course Title	**System Analysis and Design**
Credits	4
Course Objective(s)	Upon completing the System Analysis and Design course, students will be able to analyze user requirements, design effective information systems, and apply modeling techniques. They will understand the System Development Life Cycle (SDLC) and conduct feasibility studies, while also demonstrating project management skills. Additionally, students will be equipped to evaluate and test systems, preparing them for roles in system development and management
Course Outcome (CO)	**CO1**: **Understanding System Development Life Cycle (SDLC)**: Familiarize students with the phases of the SDLC, including planning, analysis, design, implementation, and maintenance. **CO2**: **Requirements Gathering**: Develop skills in techniques for gathering and analyzing user requirements to ensure that systems meet business needs. **CO3**: **Modelling Techniques**: Learn to use various modeling techniques, such as data flow diagrams (DFDs), entity-relationship diagrams (ERDs), and Unified Modeling Language (UML) to represent system processes and data structures. **CO4**: **System Design Principles**: Understand the principles of system design, including architectural design, user interface design, and database design, to create efficient and user-friendly systems. **CO5**: **Feasibility Analysis**: Conduct feasibility studies to evaluate the technical, economic, and operational viability of proposed systems. **CO6**: **Project Management**: Gain insights into project management methodologies and tools that facilitate effective planning, execution, and monitoring of system development projects. **CO7**: **Testing and Evaluation**: Learn the importance of testing and evaluation in the system development process to ensure quality and functionality.

PGDCA-205 System Analysis and Design

Unit I

System Concepts and Information Systems Environment: The System Concept: Definition, Characteristics of Systems, Elements of a
System, Open and Closed System, Formal and Informal Information Systems, Computer based Information Systems, Management
Information System, Decision Support System, General Business Knowledge, and Interpersonal Communicational System.

Unit II

System Development Life Cycle: Recognition of needs, Impetus for System Change, Feasibility Study, Analysis, Design,
Implementation, Post implementation and Maintenance. Role of the Systems Analyst, The Analyst/User Interface, Behavioural issues.
Systems planning and Initial Investigation: Strategies for Determining Information Requirement, Problem Definition & Project initiation,

Unit III

Background Analysis, Fact Analysis, Review of Written Documents, Onsite Observations, Interviews and Questionnaires, Fact Analysis,
Performance Analysis, Efficiency Analysis, Service Analysis.

Unit IV

Information Gathering: need, Information about the firms, Information gathering tools, Interviewing, Arranging the Interview, Guides to
Successful Interview, Types of Interviews and Questionnaires, The Structured and Unstructured Alternatives.

Unit V

Tools of Structured Analysis: The Dataflow Diagram (DFD), Data Dictionary, Decision Trees and Structured English.
Feasibility Study: System performance, Economic Feasibility, Technical Feasibility, Behavioral Feasibility, Steps in Feasibility Analysis.

Course Code	**PGDCA-251**
Course Title	**Web Technology Lab**
Credits	2
Course Objective(s)	Upon completing the Web Technology & Development Lab, students will be proficient in designing and developing web applications using HTML, CSS, JavaScript, and server-side technologies. They will understand web design principles, database integration, and version control, equipping them to create responsive, user-friendly websites and tackle real-world projects effectively.
Course Outcome (CO)	**CO1**: **Hands-On Experience**: Provide students with practical experience in web development technologies, including HTML, CSS, JavaScript, and server-side programming. **CO2: Web Design Principles**: Teach students the principles of effective web design, focusing on user experience (UX) and responsive design. **CO3 Frameworks and Libraries**: Familiarize students with popular web development frameworks and libraries, such as React, Angular, or Bootstrap, to enhance their development skills. **CO4**: **Version Control**: Introduce students to version control systems, such as Git, to manage code and collaborate effectively on projects. **CO5: Problem-Solving Skills**: Foster critical thinking and problem-solving skills through hands-on projects and real-world scenarios.

List of lab programs

1. What is the purpose of the **<DOCTYPE>** declaration in an HTML document?
2. How can you create a responsive layout using CSS?
3. Explain the difference between inline, block, and inline-block elements.
4. What are the different data types supported by JavaScript?
5. How do you handle events in JavaScript? Provide an example.
6. Explain the concept of closures in JavaScript.
7. What are the advantages of using a JavaScript framework like React or Angular?
8. How do you manage state in a React application?
9. What is the purpose of Bootstrap, and how does it facilitate web development?
10. What is the role of a server in web development?

Course Code	**PGDCA-252**
Course Title	**C++ Lab**
Credits	2
Course Objective(s)	Upon completing the C++ Lab, students will demonstrate proficiency in C++ programming, including object-oriented concepts, data structures, and memory management projects effectively.
Course Outcome (CO)	**CO1**: **Fundamental Concepts**: Understand and apply fundamental concepts of C++ programming, including data types, variables, operators, and control structures. **CO2: Object-Oriented Programming**: Develop proficiency in object-oriented programming principles such as classes, objects, inheritance, polymorphism, and encapsulation. **CO3: Data Structures**: Implement and manipulate basic data structures (arrays, linked lists, stacks, queue) using C++. **CO4: Memory Management**: Gain knowledge of dynamic memory allocation and deallocation using pointers and references. **CO5: File Handling**: Learn to perform file input and output operations in C++. **CO6: Standard Template Library (STL)**: Utilize the Standard Template Library to implement generic data structures and algorithms. **CO7: Debugging and Testing**: Develop skills in debugging and testing C++ programs to ensure code reliability and correctness

List of Lab programs:

1. Write a program that takes two integers as input and outputs their sum, difference, product, and quotient.
2. Create a program that checks if a number is prime or not.
3. Write a function that takes an array of integers and its size as parameters and returns the maximum value in the array.
4. Design a Rectangle class with attributes for width and height. Include methods to calculate the area and perimeter.
5. Create a base class Shape with a method to calculate area. Derive classes Circle and Square from it, implementing the area calculation for each shape.
6. Implement a program that demonstrates polymorphism using virtual functions. Create a base class Animal with a method speak(), and derive classes Dog and Cat that override the speak() method.
7. Write a program that reads a list of names from a text file and writes them to another file in reverse order.
8. Create a template function that swaps two values of any data type.
9. Write a program that prompts the user to enter a number and handles exceptions for invalid input (e.g., non-numeric input).
10. Implement a simple stack using an array or a linked list, with methods to push, pop, and display the stack contents.
11. Write a program that dynamically allocates an array of integers, fills it with user input, and then deallocates the memory.
12. Implement a sorting algorithm (e.g., bubble sort, selection sort, or quicksort) and demonstrate its use on an array of integers.
13. Write a program that implements binary search on a sorted array of integers.
14. Create a singly linked list with operations to insert, delete, and display nodes.
15. Write a simple multithreaded program that creates two threads, where one thread prints even numbers and the other prints odd numbers.

Course Code	**PGDCA-253**
Course Title	**Network Lab**
Credits	2
Course Objective(s)	Upon completing the Network Lab course, students will gain hands-on experience in configuring and troubleshooting networks, understanding protocols, and implementing security measures. They will be equipped to design, manage, and analyze networks, preparing them for real-world challenges and industry certifications.
Course Outcome (CO)	**CO1:Understand Networking Concepts and Protocols**:Understand Networking Concepts and Protocols: Gain an in-depth understanding of the fundamentals of computer networks, including layers of the OSI model and TCP/IP stack. Study common networking protocols such as IP, TCP, UDP, HTTP, DNS, and more. **CO2: Hands-on Configuration and Troubleshooting**: Learn how to configure network devices (routers, switches, firewalls, etc.) using network simulators or real devices. Develop the ability to troubleshoot network issues, understand traffic flow, and analyze network performance**.** **CO3: Network Design and Implementation**: Design, implement, and manage small-to-medium-sized networks. Understand how to set up IP addressing schemes, subnetting, and routing protocols like RIP, OSPF, and BGP. **CO4**: **Implement Network Security**: Implement basic security measures, such as setting up firewalls, VPNs, and access control lists (ACLs).Perform network security testing and configure secure communication protocols like SSL/TLS.

List of Lab Program:

1. Configure two computers with static IP addresses, connect them via a switch, and test their connectivity using the ping command. Subnet a given IP network and assign appropriate IP addresses to devices in different subnets. Verify the configuration with ping and other network utilities.
2. Set up two routers with different subnets, configure static routing between them, and test communication between devices on different subnets.
3. Configure a DHCP server on a router or dedicated machine. Set the IP address range, and ensure clients receive IP addresses automatically.
4. Configure NAT on a router to allow multiple internal devices to access the internet using a single public IP address. Verify by testing external access from internal devices.
5. Set up multiple VLANs on a switch, configure inter-VLAN routing, and test communication between devices in different VLANs.
6. Configure a basic firewall using access control lists (ACLs) to restrict traffic between two subnets and verify access control with ping or other network tools.
7. Set up a wireless router, configure SSID, security protocols (WPA2, WPA3), and test connectivity of wireless devices to the network.
8. Given a scenario with incorrect configurations or connectivity issues, use network troubleshooting tools like ping, trace route, net stat, and ipconfig to diagnose and fix the problem.
9. Use Wireshark to capture and analyze network traffic between two devices. Identify protocols and common issues like latency or dropped packets.
10. Set up RIP or OSPF on multiple routers, configure routing tables, and test network communication between different subnets.
11. Set up a site-to-site or remote-access VPN on a router or server. Test the VPN connectivity by accessing internal resources from a remote location.
12. Configure load balancing using multiple routers or switches and ensure high availability by testing failover scenarios.
13. Configure IPv6 addressing on a router and connected devices, and test IPv6 connectivity using ping and other tools.

14. Measure network performance (bandwidth, latency, jitter) using tools like iperf and analyze the results.

Course Code	**PGDCA-254**
Course Title	**Minor Project**
Credits	2
Course Objective(s)	The **Minor Project** course enables students to apply theoretical knowledge to practical problems, develop solutions, and enhance their technical and teamwork skills. By completing the project, students gain experience in project design, implementation, documentation, and effective communication, preparing them for real-world challenges.
Course Outcome (CO)	**CO1: Application of Knowledge**: To apply theoretical concepts learned during the course to solve real-world problems through the development of a functional project. **CO2: Research Skills Development**: To encourage students to conduct research, explore relevant literature, and stay updated with the latest technologies and methodologies related to their project topic. **CO3: Project Design and Implementation**: To enable students to design, develop, and implement a project based on their chosen topic, showcasing their ability to integrate various components of their field of study. **CO4: Problem-Solving and Critical Thinking**: To develop problem-solving skills by identifying issues, proposing solutions, and analyzing outcomes during the course of the project. **CO5: Technical Skills Enhancement**: To enhance students' technical skills by working on software, hardware, or systems relevant to their area of study, and applying hands-on techniques to build, test, and refine prototypes. **CO6:Teamwork and Collaboration**: To promote teamwork by encouraging collaborative work, where students contribute to various aspects of the project and learn to manage responsibilities and deadlines effectively. **CO7: Report Writing and Documentation**: To train students in preparing detailed project reports and documentation, including the ability to present their findings, methodologies, and outcomes in a structured and professional manner. **CO8:Presentation and Communication Skills**: To improve students' ability to present their projects clearly and confidently, both in written and oral formats, to peers, faculty, or external stakeholders. **CO9: Time and Resource Management**: To help students manage time and resources effectively throughout the project's lifecycle, ensuring timely completion within the constraints of the course.