

Data Structures

Code: PPCCA201

Credit: 3

Total Hours: 36

Course Objectives:

1. To provide knowledge and understanding of various basic and advanced data structures available in computing domain.
2. To provide skills to write programs to implement various data structures using procedural or object oriented programming languages.
3. To provide knowledge to analyze problems in application domains and design solution using data structures.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Be well-versed in various standard data structures available in computing domain.
2. Write programs to perform operations like searching, insertion, deletion, traversing mechanism etc. on various data structures.
3. Analyze problems; choose the appropriate data structures and write program solutions to problems in specified applications using such data structures.

Course Prerequisites:

This course requires understanding of computer programming in any language.

Detailed Syllabus:

Module 1 (10 hours)

Fundamentals: Introduction to Data Structures, Classification of Data Structures, Algorithms, Measuring Space and Time Complexities, Asymptotic Notations, Abstract Data Types.

Arrays: Storage Structures for Arrays, Sparse Matrixes, Strings, Pattern Matching.

Stacks and Queues: Representation, Operations on Stacks and Queues, Applications of Stack and Queues.

Linked Lists: Dynamic Memory Management, Single Linked Lists, Double Linked Lists, Circular Linked Lists, Linked Stacks and Queues, Operations on Polynomials.

Module 2 (10 hours)

Trees: Terminology, Representation, Binary Trees, Binary Search Trees, Searching, Insertion and Deletions Operations in a Binary Search Tree, Height Balanced Trees, M-way Search Trees, B-Trees, B+ Trees, General Trees, Representation of General Trees and Binary Trees, Forests, Application of Trees.

Module 3 (16 hours)

Graphs: Terminology, Representation, Path Matrix, Graph Traversal, Shortest Path Problems, Topological Sort.

Searching and Sorting Techniques: Linear and Binary Search, Bubble Sort, Insertion Sort, Selection Sort, Quick Sort, Merge Sort, Heap and Heap Sort, Radix Sort, Comparison of Sorting Techniques.

Hashing: Hash Functions and Hashing Techniques, External sorting.

Text Books:

1. Tremblay, Jean-Paul, and Paul G. Sorenson, "An introduction to data structures with applications", McGraw-Hill.
2. Aaron M. Tenenbaum, Data Structures Using C

Reference Books:

1. Richard F. Gilberg & Behrouz A. Forouzan, Data Structures A Psedocode Approach with C, Second Edition, CENGAGE Learning.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, "Fundamentals of Data Structures in C", Universities Press Pvt. Ltd.
3. Seymour, Lipchitz. "Data Structures with C." TMH.

Programming with Java

Code: PPCCA202**Credit: 3****Total Hours: 36****Course Objectives:**

1. To provide elaborate knowledge on standard Java language.
2. To provide knowledge on Object Oriented Approach to program design.
3. To introduce JavaFX for GUI Development

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Develop skills to write Java programs to solve a variety of real-world problems.
2. Write programs using object oriented approach and standard Java
3. Develop client server applications using network sockets
4. Develop skills to write Desktop Applications involving GUI and Databases.
5. Design programs using readable, reusable and cohesive modules.

Course Prerequisites:

This course does not require any prerequisite as such.

Detailed Syllabus:

Module 1 (10 Hours)

Features of Java, Data types, operators & expressions, control structures, arrays, Classes, objects & methods, constructors, garbage collection, access qualifiers, Overloading, String Handling – string operations, character extraction, string comparison, searching and modifying strings, String Buffer, String Builder, Packages, Interfaces, Wrapper classes, Static variables and methods.

Module 2 (14 Hours)

Inheritance: single and multilevel inheritance, method overriding, abstract class, use of super and final keywords. Exception Handling: Exception types, uncaught exceptions, multiple catch clauses, nested try statements, built-in exceptions, creating your own exceptions. Multithreading: Java thread model, creating multiple threads, thread priorities, synchronization, inter-thread communication, suspending, resuming and stopping threads; Familiarity with Java Collection Framework.

I/O Streams: Console I/O, Files I/O – Byte Streams, Character Streams, Object Serialization; Socket Programming: TCP Socket, Datagram Socket.

Module 3 (12 Hours)

JDBC programming: JDBC Drivers, Creating connection, executing queries and stored procedures, handling database transactions.

GUI Development: AWT Classes, Window fundamentals, working with graphics, working with color & fonts. Event handling in Java, Delegation Event Model, Swing Package: JFrame, JPanel, swing GUI controls, layout managers, working with menus, Introduction to JavaFX

Text Books:

1. Liang Y. Daniel, Introduction to Java Programming, Pearson Education.
2. Herbert Schildt, The Complete Reference Java 2, Tata McGraw Hill

Reference Books:

1. E. Balaguruswami, Programming with Java, Tata McGraw Hill.
2. Mughal K.A., Rasmussen R.W., A Programmer's Guide to Java Certification, Addison-Wesley

Operating Systems

Code: PPCCA203**Credit: 3****Total Hours: 36****Course Objectives:**

1. To study the main components of an OS and their functions.
2. To study the concept of process, process management and CPU scheduling algorithms.
3. To study the concepts and implementation Memory management policies and virtual memory
4. To study the file system, its implementation and disk management.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand the managerial roles of OS for resource, file system, process, memory and I/O.
2. Understand the process management policies and scheduling of processes by CPU
3. Understand process synchronization and coordination handled by operating system
4. Understand and analyze the memory management and its allocation policies.
5. Conceptualize the components involved in designing a contemporary OS.

Course Prerequisites:

This course requires understanding of computer organization and programming in any language.

Detailed Syllabus

Module 1 (10 Hours)

Operating System Introduction- Functions, Characteristics, Structures - Simple Batch, Multi programmed, timeshared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating-System services, System Calls, Virtual Machines. Process and CPU Scheduling - Process concepts and scheduling, Operation on processes, Cooperating Processes, Threads, and Interposes Communication Scheduling Criteria, Scheduling Algorithm, Multiple - Processor Scheduling, Real-Time Scheduling.

Module 2 (12 Hours)

Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors.

Deadlocks - System Model, Dead locks Characterization, Methods for Handling Deadlocks Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

Module 3 (14 Hours)

Memory Management and Virtual Memory- Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging. Demand Paging, Performance of Demanding Paging, Page Replacement, Page Replacement Algorithm, Allocation of Frames, Thrashing.

File System Interface and Implementation- Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management, Directory Management, Directory Implementation, Efficiency and Performance.

I/O Management – I/O software and its types, Disk Scheduling.

Text Books:

1. Operating System Concepts, by Abraham Silberschatz, Greg Gagne, and Peter Baer Galvin, Ninth Edition, John Wiley & Sons.

Reference Books

2. Operating Systems: Internals and Design Principles, by William Stallings, 8th edition Pearson Education Limited, 2014
3. Operating systems - A concept based Approach, by D.M Dhamdhere, 3rd Edition, Tata McGraw- Hill, 2012.
4. Operating systems, by Harvey M Deital, 3rd Edition, Pearson Education, 2011.

Database Management System

Code: PPCCA204**Credit: 3****Total Hours: 36****Course Objectives:**

1. To provide understanding on fundamental concepts of Relational database systems
2. To provide knowledge on Modeling and Design of Relational Databases
3. To provide elaborate knowledge on how to query databases
4. To provide understanding on database transactions

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand basic concepts of Relational databases management systems
2. Model data requirements of real-world applications
3. Develop databases for a variety of applications
4. Write SQL queries to perform simple to complex data manipulation tasks.

Course Prerequisites:

This course requires basic understanding of computer and programming.

Detailed Syllabus:**Module-I (10 hrs):**

Introductory concepts of DBMS: Introduction and applications of DBMS, Purpose of data base, Data, Independence, Database System architecture- levels, Mappings, Database, users and DBA

Relational Model: Structure of relational databases, Domains, Relations, Relational algebra – fundamental operators and syntax, relational algebra queries, tuple relational calculus

Entity-Relationship Model: Basic concepts, Design process, constraints, Keys, Design issues, E-R diagrams, weak entity sets, extended E-R features – generalization, specialization, aggregation, reduction to E-R database schema.

Module-II (14 hrs):

Relational Database design: Functional Dependency – definition, trivial and non-trivial FD, closure of FD set, closure of attributes, irreducible set of FD, Normalization – 1NF, 2NF, 3NF, Decomposition using FD- dependency preservation, BCNF, Multi- valued dependency, 4NF, Join dependency and 5NF.

SQL Concepts: Basics of SQL, DDL, DML, DCL, structure – creation, alteration, defining constraints – Primary key, foreign key, unique, not null, check, IN operator, Functions – aggregate functions, Built-in functions – numeric, date, string functions, set operations, sub-queries, correlated sub-queries, Use of group by, having, order by, join and its types, Exist, Any, All , view and its types; transaction control commands – Commit, Rollback, Savepoint; Introduction to PL/SQL Concepts: Cursors, Stored Procedures, Stored Function, Database Triggers

Module-III (12 hrs):

Query Processing & Query Optimization: Overview, measures of query cost, selection operation, sorting, join, evaluation of expressions, transformation of relational expressions, estimating statistics of expression results, evaluation plans, materialized views

Transaction Management: Transaction concepts, properties of transactions, serializability of transactions, testing for serializability, System recovery, Two- Phase Commit protocol, Recovery and Atomicity, Log-based recovery, concurrent executions of transactions and related problems, Locking mechanism, solution to concurrency related problems, deadlock, two-phase locking protocol, Isolation, Intent locking

Security: Introduction, Discretionary access control, Mandatory Access Control, Data Encryption

Text Books:

1. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, "Database Systems Concepts", McGraw-Hill Education , New Delhi
2. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson Education Inc., New Delhi.

Reference Books:

1. Hector Garcia-Molina, Jeffret D. Ullman, Jennifer Widom, "Database Systems: A Complete Book", Pearson Education Inc., New Delhi.
2. C. J. Date "An introduction to Database System", Pearson Education Inc., New Delhi.
3. Bipin Desai, "An introduction to Database System", Galgotia Publications.
4. Peter Rob & Carlos Coronel, "Database Systems: Design, Implementation, and Management", CENGAGE Learning India Pvt. Ltd., New Delhi.
5. Mark L. Gillenson, "Fundamentals of Database Management Systems", Wiley India Pvt. Ltd., New delhi.
6. Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", McGraw-Hill Education (India), New Delhi.

Software Engineering

Code: PPCCA205

Credit: 3

Total Hours: 36

Course Objectives:

1. To provide understanding of software process, requirement engineering and design engineering.
2. To provide knowledge on UML to develop various models for the software-to-be.
3. To provide understanding of various software testing techniques and strategies.
4. To provide understanding of quality metrics for software product.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Understand software requirements and develop SRS.
2. Analyze requirements and develop various UML models for a software-to-be at hand.
3. Display ability to test software using appropriate techniques.
4. Assess quality of software product.

Course Prerequisites:

This course requires basic understanding of computer and programming.

Detailed Syllabus:

Module-I (10 hrs):

Introduction: The evolving role of software, changing nature of software, software myths, Software engineering- a layered technology, a process framework.

Process models: The waterfall model, incremental process models, evolutionary process models, specialized process models, the unified process, agile process models.

Requirements Engineering: Requirement engineering tasks, functional and non-functional requirements, the software requirements specification (SRS), IEEE 830 guidelines, requirements validation, requirements management.

Module-II (16 hrs):

Building Analysis Models: Analysis Modeling Approaches: Structure Analysis, Object-Oriented Analysis; Data Modeling: E-R Diagram; Scenario-Based Modeling: Use-Case Model, Activity Diagram; Flow-Oriented Modeling: Data Flow Model, DFD, Class-Based Modeling: Identifying Analysis Classes, Class Model, CRC Model, Association and Dependency, Object Diagram; Behavior Modeling: State Diagram, Sequence Diagram, and Collaboration Diagram.

Building Design Models: Design process and design quality, Design Concepts, Elements of Design Model, Software Architecture: importance of software architecture, architectural styles and patterns, architectural design; Component-level Design: What is a component, Basic Design Principles, Design guidelines, Cohesion, Coupling, Component Diagram, Deployment diagram; User Interface Design: The Golden Rules, interface design evaluation cycle.

Module-III (10 hrs):

Software Testing: Verification and Validation, Unit testing and Integration Testing, Validation testing, System testing, the Art of Debugging, Black-box and White-box testing, Basis path testing, Control structure testing, Graph-Based testing, Equivalence partitioning, Boundary value analysis, Object-Oriented testing, Scenario-Based Testing, Radom testing, Partition testing, Inter-class testing, GUI testing, Client Server testing.

Product Metrics: Software quality factors, Attributes of effective software metrics, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

Text Books:

1. Roger S. Pressman, Software Engineering, A practitioner's Approach, 6th Edition, McGraw Hill International Edition.
2. Rajib Mall, Fundamentals of Software Engineering, PHI, 2014.

Reference Books:

1. I. Sommerville, Software Engineering, 9th Edition, Pearson Education.
2. Waman S. Jawadekar *Software Engineering: Principles and Practice*, Tata McGraw-Hill Education, 2004.
3. Michael R. Blaha, and James R Rumbaugh, Object-Oriented Modeling and Design with UML, 2nd Edition 2005, Pearson Education.
4. Meilir Page-Jones, Larry L. Constantine, Fundamentals of Object-oriented Design in UML, Pearson Education.

Organizational Behaviour

Code: PMCMH206

Credit: 3

Total Hours: 36

Course Objectives:

1. To develop an understanding of the behavior of individuals and groups inside organizations
2. To enhance skills in understanding and appreciating individuals, interpersonal, and group process for increased effectiveness both within and outside of organizations.
3. To develop theoretical and practical insights and problem-solving capabilities for effectively managing the organizational processes.

Course Outcomes:

On successful completion of the course, the students will be able to:

1. Compare and contrast theories of organizational behavior.
2. Analyze management issues as related to organizational behavior.
3. Assess the impact of culture on organizational behavior.
4. Understand organizational change and change management.

Course Prerequisites:

This course does not require any prerequisite.

Detailed Syllabus:

Module-I (12 hrs):

Fundamentals of OB: Definition, scope and importance of OB, Relationship between OB and the individual, Evolution of OB, Theoretical framework (cognitive), behavioristic and social cognitive), Limitations of OB.

Attitude: Importance of attitude in an organization, Right Attitude, Components of attitude, Relationship between behavior and attitude, Developing Emotional intelligence at the workplace, Job attitude, Barriers to changing attitudes.

Personality and values: Definition and importance of Personality for performance, The Myers-Briggs Type Indicator and The Big Five personality model, Significant personality traits suitable to the workplace (personality and job – fit theory), Personality Tests and their practical applications.

Perception: Meaning and concept of perception, Factors influencing perception, Selective perception, Attribution theory, Perceptual process, Social perception (stereotyping and halo effect).

Motivation: Definition & Concept of Motive & Motivation, The Content Theories of Motivation (Maslow's Need Hierarchy & Herzberg's Two Factor model Theory), The Process Theories (Vroom's expectancy Theory & Porter Lawler model), Contemporary Theories – Equity Theory of Work Motivation.

Module-II (10 hrs):

Foundations of Group Behavior: The Meaning of Group & Group behavior & Group Dynamics, Types of Groups, The Five – Stage Model of Group Development.

Managing Teams: Why Work Teams, Work Teams in Organization, Developing Work Teams, Team Effectiveness & Team Building.

Leadership: Concept of Leadership, Styles of Leadership, Trait Approach Contingency Leadership Approach, Contemporary leadership, Meaning and significance of contemporary leadership, Concept of transformations leadership, Contemporary theories of leadership, Success stories of today's Global and Indian leaders.

Module-III (14 hrs):

Organizational Culture : Meaning & Definition of Organizational Culture, creating & Sustaining Organizational Culture, Types of Culture (Strong vs. Weak Culture, Soft Vs. Hard Culture & Formal vs. Informal Culture), Creating Positive Organizational Culture, Concept of Workplace Spirituality.

Organizational Change: Meaning, Definition & Nature of Organizational Change, Types of Organizational Change, Forces that acts as stimulants to change.

Implementing Organizational Change : How to overcome the Resistance to Change, Approaches to managing Organizational Change, Kurt Lewin's-Three step model, Seven Stage model of Change & Kotter's Eight-Step plan for Implementing Change, Leading the Change Process, Facilitating Change, Dealing with Individual & Group Resistance, Intervention Strategies for Facilitating Organizational Change, Methods of Implementing Organizational Change, Developing a Learning Organization.

Text Books:

1. Understanding Organizational Behaviour, Parek, Oxford
2. Organizational Behaviour, Robbins, Judge, Sanghi, Pearson.

Reference Books:

1. Organizational Behaviour, K. Awathappa, HPH.
2. Organizational Behaviour, VSP Rao, Excel
3. Introduction to Organizational Behaviour, Moorhead, Griffin, Cengage.
4. Organizational Behaviour, Hitt, Miller, Colella, Wiley

Data Structures Lab

Code: PLCCA201

Credit: 1

Hours: 2/week

Note: This course shares the objectives and outcomes of its associated theory course PPCCA201. Suitable programming language preferably object-oriented will be used to carry out laboratory exercises. The exercises suggested below are illustrative in nature. Additional exercises may be suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Implement Stack and use it for evaluation of post-fix expression.
2. Implement conversion of prefix expression into post-fix form using recursion.
3. Implement circular queue (using array) with menu options like insert, delete, display and exit.
4. Implement a priority queue (using pointers) and use it to organize student records prioritized by marks.
5. Implement doubly linked circular list to hold strings and use it for organizing a sequence of cities constituting at our program.
6. Implement of a binary search tree with menu options: Construct a tree, insert a node, delete a node, traverse and display preorder, in order and post order sequence of its nodes.
7. Implement of di-graphs using adjacency matrix and find the transitive closure using Warshall's algorithm.
8. Implement a weighted graph and find minimal cost spanning tree using PRIM's Algorithm.
9. Generate 70 random integers in a given range and sort them using quick sort. Apply both binary search and Interpolation search to locate a given integer and compare the search algorithms based on the number of comparisons / probes required or a successful as well as unsuccessful search..
10. Implement Heap Sort, Merge Sort and other sorting algorithms on the above random numbers.
11. Implement a small Real World Application illustrating DS usage.

Programming with Java Lab**Code: PLCCA202****Credit: 2****Hours: 4 hrs/week**

Note: This course shares the objectives and outcomes of its associated theory course PPCCA202. Suitable IDE will be used to carry out laboratory exercises. The programs will follow proper object-oriented modeling. The exercises suggested below are illustrative in nature. Additionally, suitable exercises may be suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Develop an Object Oriented Program to find the area and perimeter of a circle.
2. Develop an interest calculator program to find simple interest payable monthly, compound interest payable annually compounded quarterly. Use keyboard inputs for interest rate and principal amount.
3. Define a class to calculate professional tax on a salary amount based on the following tax rate. Use if and switch control structures.

Salary Slab	Tax Rate
Up to Rs. 10000.00	Nil
Between Rs. 10001.00 – Rs. 25000.00	Rs. 100.00
Between Rs. 25001.00 – Rs. 50000.00	Rs. 200.00
Between Rs. 50001.00 – Rs. 75000.00	Rs. 300.00
Between Rs. 75001.00 – Rs. 100000.00	Rs. 450.00
Above Rs. 100000.00	Rs.650.00

4. Develop a text-menu based program to compute area and perimeter of a circle, rectangle, square and a right angle triangle.
5. Develop a menu based program to perform operations in a bank account.

6. Develop a program to find the sum of even numbers and sum of odd numbers in a set of numbers. Define a class with suitable methods to carry out the operations. Use array to store numbers.
7. Modify the class defined in sl-6 to find largest and smallest numbers in a set of numbers.
8. A student scores marks in subjects in a semester. A semester has 5 or 6 subjects depending of MCA course. Define a class called Score that contains subject code, name and marks in that subject. Define a class called Student having an array of objects of Score class in it following object composition. Process result of students in different semesters.
9. Define a class called SimpleMath with overloaded methods to carryout arithmetic operations using it. Use static methods appropriately.
10. Redefine the Circle class in exercise 1 to use value of Pi as a constant and a variable to count number of instances created as you go on creating objects.
11. Develop a class to perform the following tasks on a line of text
 - a. Count the number of words in the text
 - b. Searches a particular string in the text
 - c. Checks if the text is a palindrome
12. A library is a collection of books. Generally, a book is authored by one or more authors. Develop a program to add books and display a list of books when searched by author name. Consider title, ISBN number, publisher name, publication year for book; designation, organization, and country for author.
13. In CET, two types of people are there: students and employees. As per Govt. of India, everyone must have his or her AADHAR number for unique identification. Model these objects appropriately using inheritance and create an array of people with several students and employees in it. Write a program to search a student or an employee based on AADHAR number and print its details.
14. There are different shapes such as circles, rectangles and squares and need to be kept track of if they are painted or not with colors. Create a collection of shapes to be painted. Consider some cost of a color per square unit of the shape area. Write a program to calculate the painting cost of different shapes. Print the list of shapes which are not painted. Also print the list of painted shapes with their color, painting cost and area.
15. A bank account maintains a minimum balance. If the account balance comes down below this level due to some withdrawal then it raises warning and disallows the operation. Define a custom exception class called "InsufficientFundException" which will be raised when such event occurs. Also use the built-in exception class "IllegalArgumentException" which is to be raised when you try to either withdraw or deposit an amount less than or equal to zero.
16. Write a multithreaded program to perform following parallel operations on a set of numbers.
 - a) Find the largest number
 - b) Find the sum of the number
 - c) Sort the numbers
17. Write program using console I/O.
18. Write Programs using File I/O.
19. Write program using serialization.
20. Write client server programs using Java sockets.
21. Write JDBC programs to perform CRUD operations.
22. Write JDBC program to execute stored procedures.
23. Write GUI programs using basic swing classes
24. Write GUI program involving Menus
25. Write a GUI program using JavaFX

Operating System Laboratory

Code: PLCCA203**Credit: 1****Hours: 2 hrs/week**

Note: This course shares the objectives and outcomes of its associated theory course PPCCA203. Suitable operating system environment and programming language will be used to carry out laboratory exercises. The exercises suggested below are illustrative in nature. Additional exercises may be suggested by the faculty concerned to meet the course objectives.

LIST OF EXPERIMENTS:

1. Learn and practice basics of UNIX commands.
2. Learn and write programs using Shell Programming.
3. Implement Shared memory and IPC
4. Implement Threading and Process Synchronization Applications
5. Implement Semaphores
6. Implement the following CPU scheduling algorithms a) Round Robin b) SJF c) FCFS d) Priority
7. Implement Bankers Algorithm for Dead Lock Avoidance
8. Implement an Algorithm for Dead Lock Detection
9. Implement Paging Technique of memory management.
10. Implement page replacement algorithms: FIFO, LRU, LFU
11. Implement file allocation strategies: Sequential, Indexed, Linked
12. Implement File Organization Techniques: Single level directory, Two level, Hierarchical, DAG

Database Management System Lab

Code: PLCCA204**Credit: 1****Hours: 2/week**

This course shares the objectives and outcomes of its associated theory course PPCCA204. Suitable RDBMS such as ORACLE or MySQL will be used to carry out laboratory exercises. The exercises suggested below are illustrative in nature. Additional exercises may be suggested by the faculty concerned to meet the course objectives.

List of Exercises:

1. Create ER Models for the following: Inventory Control System, Hospital Management System, Hotel Management System, Timetable Management System, Railway Reservation System
2. Derive Database Schema from the ER Models of the above systems
3. Create a database and set constraints, relationships.
4. Perform Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions.
5. Write SQL queries using aggregate functions.
6. Write SQL queries involving sub-queries
7. Write SQL queries using group by, order by, having etc.
8. Write SQL queries involving joins.
9. Create Views, Synonyms, Sequence, Indexes, savepoint.
10. Write a PL/SQL block to satisfy some conditions by accepting input from the user.
11. Write a PL/SQL block that handles all types of exceptions.
12. Create Stored Procedures.
13. Create database triggers and functions

Software Engineering

Code: PLCCA205

Credit: 1

Hours: 2/week

This course shares the objectives and outcomes of its associated theory course PPCCA205.

Suitable UML tools will be used to carry out laboratory exercises.

List of Exercises:

1. Analyze the requirements and develop SRS for a software system. The students are to perform in groups and each group will take one application and present the SRS.
2. Draw DFD for the system. Identify various processes, data store, input, output etc. of the system and Use processes at various levels to draw the DFDs.
3. Identify the data persistence requirements and develop E-R diagram for the system
4. Identify various functions that the user will perform with the system. Develop Use-Case Model for the system.
5. Develop specification for each use-case.
6. Develop sequence diagram and collaboration diagram for each usecase following the specification.
7. Find Analysis class from the sequence diagrams and Develop class diagram and object diagram
8. Develop activity diagram and state-chart diagram for system
9. Develop component diagram and deployment diagram to show implementation view of the system
10. Use a suitable tool such as Winrunner to perform unit testing and integration testing.