M.C.A.

LOCF SYLLABUS 2023

Department of Computer Science
School of Computing Sciences
St. Joseph's College (Autonomous)
Tiruchirappalli - 620 002, Tamil Nadu, India

Vision

Forming globally competent, committed, compassionate and holistic persons, to be men and women for others, promoting a just society.

Mission

- Fostering learning environment to students of diverse background, developing their inherent skills and competencies through reflection, creation of knowledge and service.
- Nurturing comprehensive learning and best practices through innovative and value-driven pedagogy.
- Contributing significantly to Higher Education through Teaching, Learning, Research and Extension.

Programme Educational Objectives (PEOs)

- Graduates will be able to accomplish professional standards in the global environment.
- Graduates will be able to uphold integrity and human values.
- Graduates will be able to appreciate and promote pluralism and multiculturalism in working environment.

Programme Outcomes (POs)

1. Graduates will be able to apply assimilated knowledge to evolve tangible solution to emerging problems.
2. Graduates will be able to analyze and interpret data to create and design new knowledge.
3. Graduates will be able to engage in innovative and socially relevant research and effectively communicate the findings.
4. Graduates will become ethically committed professional and entrepreneurs upholding human values.
5. Graduates imbibed with ethical values and social concern will be able to understand and appreciate cultural diversity, social harmony and ensure sustainable environment.

Programme Specific Objectives (PSOs)

1. Graduates will be able to implement the logic for solving the real life problems by using the knowledge gained
2. Graduates will be able to understand, analyze, design, develop, test, implement and document software systems
3. Graduates will be able to use their creative skill to evolve new ideas, defend their findings at the peer level and able to manage IT and ITES organizations.
4. Graduates will be able to work in public and private sectors satisfying social and environmental obligations with multiple cultures.
5. Graduates will be able to act as socially responsible IT professionals or service minded entrepreneurs.

CONTINUOUS INTERNAL ASSESSMENT
Categorizing Outcome Assessment Levels Using Bloom's Taxonomy

Level	Cognitive Domain	Description
K1	Remember	It is the ability to remember the previously learned concepts or ideas.
K2	Understand	The learner explains concepts or ideas.
K3	Apply	The learner uses existing knowledge in new contexts.
K4	Analyse	The learner is expected to draw relations among ideas and to compare and contrast.
K5	Evaluate	The learner makes judgements based on sound analysis.
K6	Create	The learner creates something unique or original.

Question Paper Blueprint for Mid and End Semester Tests

<table>
<tr><td colspan="3">Duration: 2 Hours</td><td colspan="6">Maximum Marks: 60</td></tr>
<tr><td colspan="2" rowspan="2">Section</td><td colspan="6">K level*</td><td rowspan="2">Marks</td></tr>
<tr><td>K1</td><td>K2</td><td>K3</td><td>K4</td><td>K5</td><td>K6</td></tr>
<tr><td colspan="2">A (no choice)</td><td>7</td><td></td><td></td><td></td><td></td><td></td><td>7 × 1 = 7</td></tr>
<tr><td colspan="2">B (no choice)</td><td></td><td>5</td><td></td><td></td><td></td><td></td><td>5 × 3 = 15</td></tr>
<tr><td colspan="2">C (either… or type)</td><td></td><td></td><td>3</td><td></td><td></td><td></td><td>3 × 6 = 18</td></tr>
<tr><td rowspan="4">D (2 out of 3)</td><td>Courses with K4 as the highest cognitive level</td><td></td><td></td><td></td><td>2</td><td></td><td></td><td rowspan="4">2 × 10 = 20</td></tr>
<tr><td>Courses with K5 as the highest cognitive level wherein one question each on K4 and K5 is compulsory.
(Note:K4 has two questions whereas, K5 has no choice.)</td><td></td><td></td><td></td><td>1</td><td>1</td><td></td></tr>
<tr><td rowspan="2">Courses with K6 as the highest cognitive level wherein one question each on K5 and K6 is compulsory.
(Note: Mid Sem: K4 has two questions whereas, K5 has no choice; End sem: K5 has two questions whereas, K6 has no choice)</td><td></td><td></td><td></td><td colspan="2">Mid Sem</td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2">End Sem</td></tr>
<tr><td colspan="6">Total</td><td>60</td></tr>
</table>

** K4 and K5 levels will be assessed in the Mid semester test whereas K5 and K6 levels will be assessed in the End semester test.*

Question Paper Blueprint for Mid and End Semester Tests *(For quantitative courses only)*

Duration: 2 Hours							**Maximum Marks: 60**
Section	**K level**						**Marks**
	K1	**K2**	**K3**	**K4**	**K5**	**K6**	
A *(no choice)*	5	4					9 × 1 = 9
B *(either… or type)*			2	1			3 × 5 = 15
C *(2 out of 3)*					1	1*	2 × 18 = 36
						Total	**60**

NOTE: *K4 and K5 will be assessed in the Mid semester test whereas K5 and K6 will be assessed in the End semester test.*
** K6 compulsory*

SEMESTER EXAMINATION
Question Paper Blueprint for Semester Examination

Duration: 3 Hours								**Maximum Marks: 100**
Section		**K level**						**Marks**
		K1	**K2**	**K3**	**K4**	**K5**	**K6**	
A *(no choice, two questions from each unit)*		10						10 × 1 = 10
B *(no choice, two questions from each unit)*			10					10 × 3 = 30
C *(either… or type, one question from each unit)*				5				5 × 6 = 30
D *(3 out of 5, one question from each unit)*	*Courses with K4 as the highest cognitive level*				3			3 × 10 = 30
	Courses with K5 as the highest cognitive level wherein two K4 questions and one K5 question are compulsory. (Note: Three questions on K4 and two questions on K5)				2	1		
	Courses with K6 as the highest cognitive level wherein one question each on K4, K5, and K6 is compulsory. (Note: Two questions each on K4 and K5 and one question on K6)				1	1	1	
							Total	**100**

Question Paper Blueprint for Semester Examination *(For quantitative courses only)*

Section	Marks	K level
A	10 × 1 = 10	K1
B	5 × 6 = 30 *(either…or)*	K2 *(Q. No. 11 & 12)* K3 *(Q. No. 13, 14 & 15)*
C	4 × 15 = 60 *(4 out of 5)*	K4 *(Q. No. 16 & 17)* K5 *(Q. No. 18 & 19)* K6 *(Q. No. 20 compulsory)*
Total Marks: 100		

Evaluation Pattern for Part IV One/Two Credit Courses

Title of the Course	CIA	Semester Examination	Total Marks
Internship	100		**100**
UG Skill Enhancement Course (Non Major Elective) Foundation Course **PG** Ability Enhancement Course	20 + 10 + 20 = 50	50 *(External member from the Department)*	**100**
Value Education	50	50 *(CoE)*	**100**

MCA							
PROGRAMME PATTERN							
Course Details					**Scheme of Exams**		
Sem	**Course Code**	**Title of the Course**	**Hours**	**Credits**	**CIA**	**SE**	**Final**
1	23PCA1CC01	**Core Course - 1:** C++ and Data Structures	4	4	100	100	100
	23PCA1CC02	**Core Course - 2:** Introduction to Computer Architecture	4	4	100	100	100
	23PCA1CC03	**Core Course - 3:** Relational Database Management Systems	4	4	100	100	100
	23PCA1CP01	**Core Practical - 1:** Data Structures using C++	4	2	100	100	100
	23PCA1CP02	**Core Practical - 2:** RDBMS	4	2	100	100	100
	23PCA1ES01	**Elective - 1:** Accounting and Financial Management	4	3	100	100	100
	23PCA1ES02	**Elective - 2:** Theory of Computation	4	3	100	100	100
	23PCA1AE01	**Ability Enhancement Course**: Programming in Java	2	1	100	-	100
		Total	**30**	**23**			
2	23PCA2CC04	**Core Course - 4:** Programming Smart Devices	4	4	100	100	100
	23PCA2CC05	**Core Course - 5:** Software Engineering	4	3	100	100	100
	23PCA2CC06	**Core Course - 6:** Data Analysis Using Python	4	3	100	100	100
	23PCA2CP03	**Core Practical - 3:** Programming Smart Devices	3	3	100	100	100
	23PCA2CP04	**Core Practical - 4:** Python Programming	2	2	100	100	100
	23PCA2SP01	**Self-paced Learning**: XML*	-	2	50	50	50
	23PCA2ES03A	**Elective - 3:** Internet of Things	5	4	100	100	100
	23PCA2ES03B	**Elective - 3**: Cloud Computing					
	23PSS2SE01	**Skill Enhancement Course:** Soft Skills	4	3	100	-	100
	23PCA2EG01	**Generic Elective - 1: (WS)** Applied Statistics Using R	4	3	100	100	100
	-	Extra Credit Course (MOOC/Certificate Course) - 1	-	(3)			
		Total	**30**	**27 (3)**			
3	23PCA3CC07	**Core Course - 7:** Distributed Technologies	5	5	100	100	100
	23PCA3CC08	**Core Course - 8:** Computer Networks and Security	5	5	100	100	100
	23PCA3CC09	**Core Course - 9:** Operations Research	5	4	100	100	100
	23PCA3CP05	**Core Practical - 5:** Distributed Technologies	3	3	100	100	100
	23PCA3CP06	**Core Practical - 6:** Web App Development using MEAN	3	2	100	100	100
	23SCS3CC01	**Common Core:** Design and Analysis of Algorithms	5	4	100	100	100
	23PCA3EG02	**Generic Elective - 2 (BS):** Web Design	4	3	100	100	100
	23PCA3IS01	Internship	-	2	100	-	100
	-	Extra Credit Course (MOOC/Certificate Course) - 2		(3)			
		Total	**30**	**28 (3)**			
4	23PCA4PW01	Project Work and Viva Voce	25	22	100	100	100
	23PCA4CE01	Comprehensive Examination*	-	2	50	50	50
	23PCA4ES04A	**Elective - 4:** Recent Trends in Computer Science - 1 #	5	4	100	100	100
	23PCA4ES04B	**Elective - 4:** Recent Trends in Computer Science - 2 #					
	-	Extra Credit Course (MOOC / Certificate Course) - 3	-	(3)			
		Total	**30**	**28(3)**			
2 - 4	23PCW4OR01	Outreach Programme	-	4			
1 - 4		**Total (Four Semesters)**	**120**	**110**			

23PCA1BC01	Bridge Course**	30	2

*- for grade calculation 50 marks are converted into 100 in the mark statements

- Blended Learning (online and offline)- The students can opt any of the Elective - 4.

** - Mandatory Bridge Course for all Non-Computer Science Stream Students. Two weeks to be conducted outside the class hours and evaluated for 100 marks (Purely Internal).

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCA1CC01	**Core Course - 1:** C++ and Data Structures	4	4

Course Objectives
To develop a solid understanding of the fundamental concepts of C++ programming
To gain in object-oriented programming by comprehending the concepts of classes, objects, constructors, and inheritance
To acquire the skills necessary to handle file operations, including opening, closing, updating, and error handling
To master the implementation and application of stack data structure, including infix to postfix conversion, recursion
To attain a thorough understanding of tree and graph data structures, including binary trees, traversals, and graphs

UNIT I: Introduction to C++ (12 hours)

Tokens, Keywords, Identifiers, Variables, Operators, Manipulators, Expressions and Control Structures in C++; Pointers - Functions in C++ - Main Function -Function Prototyping - Parameters Passing in Functions - Values Return by Functions - Inline Functions - Friend and Virtual Functions

UNIT II: Classes and Objects (12 hours)

Constructors and Destructors; and Operator Overloading and Type Conversions - Type of Constructors - Function overloading. Inheritance: Single Inheritance - Multilevel Inheritance - Multiple Inheritance - Hierarchical Inheritance - Hybrid Inheritance. Pointers, Virtual Functions and Polymorphism; Managing Console I/O operations.

UNIT III: Working with Files (10 hours)

Classes for File Stream Operations - Opening and Closing a File – End of File Deduction - File Pointers - Updating a File - Error Handling during File Operations – Command line Arguments.

UNIT IV: Stack (14 hours)

Data Structures: Definition of a Data structure - primitive and composite Data Types, Asymptotic notations, Arrays, Operations on Arrays,Order lists. Applications of Stack - Infix to Postfix Conversion, Recursion, Maze Problems - Queues- Operations on Queues, Queue Applications, Circular Queue. Singly Linked List- Operations, Application - Representation of a Polynomial, Polynomial Addition; Doubly Linked List - Operations, Applications.

UNIT V: Trees and Graphs (12 hours)

Binary Trees - Conversion of Forest to Binary Tree, Operations - Tree Traversals; Graph - Definition, Types of Graphs, Hashing Tables and Hashing Functions, Traversal - Shortest Path; Dijkstra's Algorithm.

Teaching Methodology	Lecture-based instruction, Demonstration, Group Discussion, Peer Learning, Problems solving, and Project-based learning

Books for Study

1. Horowitz, E., Sahni. S. & Mehta. (2008). *Fundamentals of data structures in C++* (2nd ed.). Galgotia.
2. Schildt, H. (1999). *C++ - The complete reference* (3rd ed.). Tata McGraw – Hill.
3. Goodrich, M. T., Tamassia, R. & Mount, D. M. (2007). *Data structures and algorithms in C++*. Wiley.

Books for Reference

1. Heileman, G. L. (1996), *Data structures, algorithms and object oriented programming*. Mc-Graw Hill International Editions.
2. Aho, A. V., Ullman, J. D. & Hopcraft, J. E. (1974), *Data structures and algorithms*. Adisson Wesley Publication.
3. Salaria, R. S. (2018). *Data structures and algorithms Using C++*. Kanna Book Publishing.

Web Resources

1. https://www.geeksforgeeks.org/data-structures/
2. https://www.tutorialspoint.com/cplusplus/cpp_data_structures.htm
3. https://www.programiz.com/cpp-programming/data-structure
4. https://www.codecademy.com/learn/learn-c-plus-plus/modules/learn-cpp-data-structures
5. https://cslibrary.stanford.edu/110/BinaryTrees.html

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to,	**Cognitive Levels (K - Level)**
CO1	define the fundamental concepts of C++ programming	**K1**
CO2	summarize the principles of object-oriented programming.	**K2**
CO3	apply different techniques of C++ to create and manipulate files.	**K3**
CO4	analyze the applications of data structure and develop programs using the data structure	**K4**
CO5	build and manipulate different data structure in C++, applying the same to develop algorithms.	**K5**
CO6	design and implement complex programs that involve multiple concepts, data structures, and algorithms.	**K6**

Relationship Matrix											
Semester	**Course code**		**Title of the Course**							**Hours**	**Credits**
1	**23PCA1CC01**		**Core Course - 1:** C++ and Data Structures							**4**	**4**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**2**	**2**	**1**	**2**	**2**	**3**	**2**	**3**	**3**	**3**	**2.3**
CO2	**2**	**2**	**1**	**2**	**3**	**2**	**2**	**3**	**2**	**3**	**2.2**
CO3	**2**	**2**	**1**	**2**	**3**	**3**	**2**	**2**	**3**	**2**	**2.2**
CO4	**1**	**2**	**2**	**2**	**3**	**2**	**2**	**3**	**2**	**3**	**2.2**
CO5	**2**	**2**	**3**	**3**	**2**	**2**	**3**	**2**	**3**	**3**	**2.5**
CO6	**2**	**2**	**3**	**2**	**2**	**2**	**2**	**3**	**3**	**3**	**2.4**
Mean overall Score											**2.3 (High)**

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCA1CC02	**Core Course - 2:** Introduction to Computer Architecture	4	3

Course Objectives
Understand the Digital number system and their conversions
Identify the operations of logic Gates and simplify the Boolean expressions using K-Map
Comprehend the fundamental principles of simple Arithmetic Circuits
Realize the design of sequential logic circuits such as Flip Flops, Registers and Counters and its applications
Gain the knowledge about the memory elements like RAM, ROM, and Magnetic Disk memories and Secondary Memories.

UNIT I: Digital and Number System (12 Hours)

Data and Information Features of Digital Systems, Number Systems. Decimal, Binary, Octal, Hexadecimal and their inter conversions, Representation of Data: Signed Magnitude, one's complement and two's complement, Binary Arithmetic, Fixed point representation and Floating-point representation of numbers. Codes BCD, XS-3, Gray code, hamming code, alphanumeric codes (ASCII, EBCDIC, UNICODE), Error detecting and error correcting codes

UNIT II: Boolean Algebra (12 Hours)

Boolean Algebra: Basic gates (AND, OR, NOT gates), Universal gates (NAND and NOR gates), other gates (XOR, XNOR gates). Boolean identities, De Morgan Laws. Karnaugh maps: SOP and POS forms, Quine McClusky method.

UNIT III: Combinational Circuits (12 Hours)

Combinational Circuits: Half adder, full adder, code converters, combinational circuit design, Multiplexers and demultiplexers, encoders, decoders, Combinational design using mux and demux, PLA.

UNIT IV: Sequential Circuit Design (12 Hours)

Sequential Circuit Design: Flip flops RS, Clocked RS, D, JK, JK Master Slave, T, Counters, Shift registers and their types, Counters: Synchronous and Asynchronous counters.

UNIT- V: ALU Structure & Memory (12 Hours)

ALU Structure – Memory: ROM, RAM, PROM, EPROM, EEPROM, and Secondary Memory: Hard Disk and optical Disk, Cache Memory, I/O devices.

Teaching Methodology	Videos, PPT, Demonstration, and Designing Logic Circuit

Books for Study

1. Jain, R. P. (2008). *Modern digital electronics*. McGraw Hill.
2. Gill, N. S. & Dixit, J. B. (2016). *Digital design and computer organization*.University Science Press, Sausalito, CA, United States.
3. Norton, P. (2005). *Introduction to computers*. McGraw Hill.

Books for Reference

1. Malvino & Leach (2014). *Digital principles and applications*. McGraw Hill, New York.
2. Balagurusamy (2009). *Introduction to computers*. McGraw Hill Education, New York.

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K - Level)**
CO1	recall the fundamentals of digital logic and elements of a digital computer	**K1**
CO2	demonstrate the logics of sequential and combinational circuits	**K2**
CO3	solve the problems on logic circuits using digital logics	**K3**
CO4	classify the digital logics of sequential and combinational circuits	**K4**
CO5	interpret the functioning of logic circuits and memory elements	**K5**
CO6	design digital circuit based on the given constraints	**K6**

Relationship Matrix

Semester	Course code	Title of the Course					Hours	Credits			
1	23PCA1CC02	Core Course - 2: Introduction to Computer Architecture					4	3			
Course Outcomes	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Score of COs
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	3	3	2	2	2	3	2	3	2	3	2.5
CO2	3	3	3	2	1	3	3	3	2	2	2.5
CO3	2	3	3	2	1	2	3	3	2	2	2.3
CO4	3	3	3	2	1	3	3	3	2	2	2.5
CO5	3	3	3	1	1	2	3	3	2	2	2.3
CO6	3	3	3	2	3	3	2	3	2	3	2.7
Mean overall Score											2.56 (High)

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCA1CC03	**Core Course - 3:** Relational Database Management Systems	4	3

Course Objectives
To learn the fundamentals of data models, SQL and to represent a database system using ER Diagrams
To study relational database design concepts and normalization procedures
To understand the fundamental concepts of transaction processing, concurrency control techniques and recovery procedures
To understand the internal storage structure using different file systems and indexing techniques which will help in physical database design
To gain fundamental knowledge on other databases like Distributed, XML and Object-relational databases

UNIT I: Relational Databases **(12 Hours)**

Purpose of Database System – Views of data – Data Models – Database System Architecture – Introduction to relational databases – Relational Model – Keys – Relational Algebra – SQL fundamentals – Advanced SQLfeatures – Embedded SQL– Dynamic SQL.

UNIT II: Database Design **(12 Hours)**

Entity-Relationship model – E-R Diagrams – Enhanced- ER Model – ER-to-Relational Mapping – Functional Dependencies – Non-loss Decomposition – First, Second, Third Normal Forms, Dependency Preservation – Boyce/Codd Normal Form – Multi-valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form

UNIT III: Transactions **(12 Hours)**

Transaction Concepts – ACID Properties – Schedules – Serializability – Concurrency Control – Need for Concurrency – Locking Protocols – Two Phase Locking – Deadlock – Transaction Recovery - Save Points – Isolation Levels – *SQL Facilities for Concurrency and Recovery.*

UNIT IV: Implementation Techniques RAID **(12 Hours)**

File Organization – Organization of Records in Files – Indexing and Hashing –Ordered Indices – B+ tree Index Files – B tree Index Files – Static Hashing – Dynamic Hashing – *Query Processing Overview – Algorithms for SELECT and JOIN operations – Query optimization using Heuristics and Cost Estimation.*

UNIT V: Advanced Topics **(12 Hours)**

Distributed Databases: Architecture, Data Storage, Transaction Processing – Object-based Databases: Object Database Concepts, Object- Relational features, ODMG Object Model, ODL, OQL - XML Databases: XML Hierarchical Model, DTD, XML Schema, XQuery – *Information Retrieval: IR Concepts, Retrieval Models, Queries in IR systems.*

Teaching Methodology	a) Provide Exercises for SQL Queries, Data Modeling and Normalization b) Assign group work to design relational databases c) Conduct regular quizzes to evaluate the knowledge level of the students d) Provide students with relevant OER references

Books for Study

1. Silberschatz, A., Korth, H. F. & Sudharsha. S. (2011). *Database System Concepts* (6th ed.). Tata McGraw Hill.
2. Elmasri, R., Navathe, S. B. (2011). *Fundamentals of Database Systems* (6th ed.). Pearson Education.

Books for Reference

1. Date, C. J., Kannan, A. & Swamynathan. S. (2006). *An Introduction to Database Systems* (8th ed.). Pearson Education.
2. Ramakrishnan, R. (2015). *Database Management Systems* (4th ed.). McGraw Hill, College Publications.
3. Gupta, G. K. (2011). *Database Management Systems*. Tata McGraw Hill.

Websites and eLearning Sources

1. https://www.w3schools.com/sql/
2. https://www.studytonight.com/dbms/database-normalization.php
3. https://www.databasejournal.com/

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K - Level)**
CO1	recall the key concepts and terminologies in relational and advanced database systems	**K1**
CO2	interpret the implementation scenarios of database design transactions and storage mechanisms in relational data model	**K2**
CO3	map ER Model to relational model, normalize data and formulate SQL queries	**K3**
CO4	classify data accessing strategies in different types of database systems	**K4**
CO5	appraise how advanced databases differ from traditional databases	**K5**
CO6	build a complete relational database design with proper normalizations	**K6**

Relationship Matrix											
Semester	**Course code**	**Title of the Course**								**Hours**	**Credits**
1	**23PCA1CC03**	**Core Course - 3:** Relational Database Management Systems								**4**	**3**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	3	2	3	1	2	2	2	3	2	2	2.2
CO2	3	3	2	1	1	3	2	3	2	2	2.2
CO3	3	2	3	1	1	3	3	3	2	2	2.3
CO4	2	3	3	1	3	1	3	2	2	3	2.3
CO5	3	2	3	2	2	2	2	3	1	2	2.2
CO6	3	3	3	2	3	3	2	3	2	3	2.4
Mean overall Score											**2.26 (High)**

Semester	Course Code	Title of the Course	Hours	Credits
1	**23PCA1CP01**	**Core Practical - 1:** Data Structures using C++	**4**	**2**

C++

1. Class and Objects
2. Functions
3. Constructors
4. Inheritance
5. Pointers
6. File Handling

Data Structures

7. Array
8. Stack and Queue
9. Linked List
10. Binary Tree Traversals

Semester	Course Code	Title of the Course	Hours	Credits
1	**23PCA1CP02**	**Core Practical - 2:** RDBMS	**4**	**2**

SQL

1. DDL, DML and DCL Queries
2. Set Operations
3. Views
4. Joins
5. Sub Queries
6. Indexes, Sequence and Synonyms

PL/SQL

7. Cursors
8. Functions and Procedures
9. Packages
10. Triggers

FORMS AND REPORTS

11. Forms – Menus, Buttons, LOVs, Master-Detail form design
12. Simple Report Design

Semester	Course Code	Title of the Course	Hours	Credits
1	**23PCA1ES01**	**Elective - 1:** Accounting and Financial Management	4	3

Course Objectives
To understand the fundamental principles of accounting
To develop the ability to analyze and interpret financial statements
To critically analyze and provide recommendations to improve the operations of organizations
To acquire the skills to prepare functional budgets and understand their characteristics.
To develop a comprehensive understanding of project appraisal techniques

UNIT I: Introduction to the Principles of Accounting (12 hours)

Principles of double entry -Assets and Liabilities - Accounting records and systems - Trial balance and preparation of financial statements - Trading, Manufacturing, Profit and Loss accounts, Balance Sheet including adjustments (Simple problems only).

UNIT II: Analysis and Interpreting Accounts and Financial Statements (12 hours)

Ratio analysis - Use of ratios in interpreting the final accounts (trading accounts and loss a/c and balance sheet) - final accounts to ratios as well as ratios to final accounts.

UNIT III: Break-even analysis and Marginal Costing (12 hours)

Meaning of variable cost and fixed cost – Cost-Volume -Profit analysis – calculation of breakeven point, Profit planning, sales planning and other decision – making analysis involving break - even analysis - Computer Accounting and algorithm.(differential cost analysis to be omitted)

UNIT IV: Budget/Forecasting (12 hours)

Preparation of and Characteristics of functional budgets, Production, sales, Purchases, cash and flexible budgets.

UNIT V: Project Appraisal (12 hours)

Method of capital investment decision making: Payback method , ARR method - Discounted cash flows - Net Present values - Internal rate of return - Sensitivity analysis - Cost of capital

Teaching Methodology	Lecture-based Teaching, Case-Studies and Problem Solving, Problem-based learning

Books for Study

1. Shukla, M. C. & Grewal, T. S. (1991). *Advanced Accounts*, S. Chand & Co.
2. Gupta R.L. & Radhaswamy, M. (1991). *Advanced Accounts* Vol. II, Sultan Chand & Sons.

3. Maheswari, S. N. (2021). *Principles of Management Accounting*. Sultan Chand.
4. Ramachandran, R. & Srinivasan, S. (2017). *Management Accounting (Theories, Problems & Solutions)* (6th ed.). Sriram Publications.

Books for Reference

1. Kuchhal, S. C. (1980). *Financial Management*. Chaitanya.
2. Mohan, M. & Goyal, S. N. (1987). *Principles of Management Accounting*. Arya Sahithya Bhawan.
3. Hingorani, N. L. & Ramanthan, A. R. (1992). *Management Accounting* (5th ed.). Sultan Chand.

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K - Level)**
CO1	recall and comprehend the fundamental principles, concepts, and terminology of accounting	**K1**
CO2	explain the purpose and significance of financial statements in business decision-making	**K2**
CO3	solve accounting problems and make informed decisions based on financial data and analysis	**K3**
CO4	compare the findings from financial analysis and provide insights and recommendations for management	**K4**
CO5	assess the effectiveness of budgeting and forecasting in planning and controlling financial activities	**K5**
CO6	elaborate the importance of ethical considerations in accounting practices and decision-making	**K6**

Relationship Matrix

Semester	**Course code**	**Title of the Course**					**Hours**	**Credits**
1	23PCA1ES01	**Elective - 1:** Accounting and Financial Management					4	3

Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**3**	**2**	**2**	**3**	**2**	**3**	**2**	**2**	**2**	**1**	**2.2**
CO2	**3**	**3**	**1**	**3**	**2**	**2**	**3**	**2**	**2**	**1**	**2.2**
CO3	**1**	**2**	**3**	**2**	**2**	**2**	**3**	**2**	**3**	**2**	**2.2**
CO4	**3**	**3**	**1**	**2**	**1**	**1**	**2**	**3**	**2**	**3**	**2.1**
CO5	**2**	**3**	**2**	**3**	**3**	**3**	**2**	**2**	**2**	**2**	**2.4**
CO6	**2**	**3**	**2**	**3**	**3**	**3**	**2**	**2**	**2**	**2**	**2.4**
										Mean overall Score	**2.3 (High)**

Semester	Course Code	Title of the Course	Hours	Credits
1	23PCA1ES02	**Elective - 2:** Theory of Computation	4	3

Course Objectives
To give an overview of the theoretical foundations of computer science from the perspective of formal languages
To illustrate finite state machines to solve problems in computing
To explain the hierarchy of problems arising in theory of computation
To familiarize various types of gramma
To use basic concepts of formal languages and finite automata techniques

UNIT I: Review of Mathematical Theory (12 Hours)

COMBINATORICS Review of Permutation and Combination - Mathematical Induction - Pigeon hole principle - Principle of Inclusion and Exclusion - generating function - Recurrence relations. Statements – Connectives – Truth Tables – Normal forms – Predicate calculus – Inference – Theory for Statement Calculus and Predicate Calculus

UNIT II: Regular Languages and Finite Automata (12 Hours)

Regular Expressions, Regular Languages, Application of Finite Automata, Automata with output - Moore machine & Mealy machine, Finite Automata, Memory requirement in a recognizer, Definitions, union- intersection and complement of regular languages, Non-Deterministic Finite Automata, Conversion from NFA to FA- Non-Deterministic Finite Automata, Conversion of NFA- to NFA, Kleene's Theorem, Minimization of Finite automata, Regular And Non Regular Languages – pumping lemma.

UNIT III: Context free grammar (CFG) (12 Hours)

Definitions and Examples, Unions Concatenations and Kleene's of Context free language, Regular Grammar for Regular Language, Derivations and Ambiguity, Unambiguous CFG and Algebraic Expressions, Backaus Naur Form (BNF), Normal Form – CNF.

UNIT IV: Pushdown Automata, CFL and NCFL (12 Hours)

Definitions, Deterministic PDA, Equivalence of CFG and PDA & Conversion, Pumping lemma for CFL, Intersections and Complements of CFL, Non-CFL.

UNIT V: Turing Machine (TM) (12 Hours)

TM Definition, Model of Computation, Turing Machine as Language Acceptor, TM that Compute Partial Function, Church Turing Thesis, Combining TM, Variations Of TM, Non-Deterministic TM, Universal TM, Recursively and Enumerable Languages, Context sensitive languages and Chomsky hierarchy.

Note: Emphasis is given only on basic concepts and problems (No Proof and Derivations)

Teaching Methodology	Chalk and Talk, Videos, PPTs, Group Discussion and Problem solving

Books for Study

1. Tremblay, J. P. & Manohar, R. (1997). *Discrete Mathematical Structures with Applications to Computer Science*. TATA McGraw-Hill Edition.
2. Hopcroft, J. E. & Ullman, J. D. (1979). *Introduction to Automata Theory, Languages and Computation*. Narosa Publishing House.
3. Linz, P. (2016). *An Introduction to Formal Languages and Automat* (6th ed.). Jones & Bartlett Learning.

Books for Reference

1. Mishra, K. L. P. & Chandrashekaran, N. (2003). T*heory of Computer Science- Automata Languages and Computation* (2nd ed.). Prentice Hall.
2. Hopcroft, J. E., Motwani, R. & Ullman, J. D. (2007). *Introduction to Automata Theory Languages and Computation* (3rd ed.). Pearson Education.

Web References

1. https://nptel.ac.in/courses/106106049

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K - Level)**
CO1	familiarize with the basics of Theory of Computation	**K1**
CO2	apply the principles of languages and finite automata to solve problems related to regular expressions, regular languages, finite automata with output, memory requirements, and operations on languages	**K2**
CO3	demonstrate proficiency in grammars by defining and constructing grammars for various languages, understanding the concepts of ambiguity, unambiguity, and normal forms	**K3**
CO4	understand the concept and functionality of machines as a model of computation, including language acceptance, computation of partial functions	**K4**
CO5	apply critical thinking and problem-solving skills to analyze complex computational problems and devise appropriate solutions using the concepts learned in theory of computation	**K5**
CO6	demonstrate advanced knowledge and understanding of theoretical aspects of computation, including advanced topics such as advanced combinatorics, advanced formal languages, complexity theory, and computability theory	**K6**

Relationship Matrix											
Semester	**Course code**		**Title of the Course**							**Hours**	**Credits**
1	**23PCA1ES02**		**Elective - 2:** Theory of Computation							**4**	**3**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**2**	**3**	**2**	**3**	**2**	**2**	**3**	**2**	**1**	**3**	**2.3**
CO2	**1**	**2**	**2**	**3**	**2**	**2**	**2**	**3**	**2**	**3**	**2.2**
CO3	**2**	**2**	**3**	**3**	**3**	**1**	**3**	**3**	**3**	**2**	**2.5**
CO4	**2**	**3**	**3**	**2**	**2**	**3**	**3**	**2**	**3**	**2**	**2.5**
CO5	**1**	**2**	**2**	**2**	**3**	**2**	**3**	**3**	**3**	**3**	**2.4**
CO6	**1**	**2**	**2**	**3**	**2**	**2**	**3**	**2**	**2**	**3**	**2.2**
Mean overall Score											**2.35 (High)**

Semester	Course Code	Title of the Course	Hours	Credits
1	23PCA1AE01	**Ability Enhancement Course:** Programming in Java	2	1

Course Objectives
To understand the fundamentals of Object-Oriented Programming
To familiar with the syntax and structure of Java programming
To explore the different data types and operators in Java
To understand the significance of decision-making statements in Java programming
To gain knowledge about classes and objects in Java

UNIT I: Introduction to OOPS (6 hours)

Paradigms of Programming Languages –Basic concepts of Object Oriented Programming – Differences between Procedure Oriented Programming and Object Oriented programming - Benefits of OOPs – Application of OOPs.

UNIT II: Introduction to Java (6 hours)

History – Java features – Java Environment – JDK – API. Introduction to Java: Types of java program – Creating and Executing a Java program – Java Tokens- Java Virtual Machine (JVM) – Command Line Arguments –Comments in Java program.

UNIT III: Data types and Operators (6 hours)

Constants – Variables – Data types - Scope of variables – Type casting – Operators: Special operators –Expressions – Evaluation of Expressions.

UNIT IV: Looping Statements and Arrays (6 hours)

Decision making and branching statements- Decision making and Looping– break – continue statement- Arrays: One Dimensional Array – Multidimensional Array.

UNIT V: Class and objects (6 hours)

Defining a class – Methods – Creating objects– Accessing class members – Constructors – Method overloading – Static members – this keyword – Inheritance: Defining inheritance – types of inheritance – JDBC Connectivity.

List of Practical's

1. Write a Java program to find area and perimeter of circle.
2. Write a java Program to find factorial of a given number.
3. Write a java program to find simple and compound Interest
4. Write a Java program to find sum of n numbers using array
5. Write a simple Java program using class & objects.

Teaching Methodology	PPT, Demonstration

Books for Study

1. Balagurusamy, E. (2014). *Programming with Java* (5th ed.). Tata McGraw Hill Education (India) Private Limited.
2. Sagayaraj et al. (2018). *Java Programming for Core andAdvanced Learners*. Universities Press (India) Private Limited.

Books for Reference

1. Schildt, H. (2007). *The complete reference Java* (7th ed.). Tata McGraw Hill Education (India) Private Limited.
2. Muthu, C. (2011). *Programming with Java* (2nd ed.). Vijay Nicole Imprints Private Limited.

Web Resources

1. https://www.javatpoint.com/java-tutorial
2. https://www.geeksforgeeks.org/java/

Course Outcomes		
CO No.	**CO-Statements**	**Cognitive Levels (K - Level)**
	On successful completion of this course, students will be able to	
CO1	recall the fundamentals concepts in java programming.	**K4**
CO2	understand the different types of inheritance.	**K5**
CO3	apply the object-oriented programming concepts to write simple java programs.	**K6**

Relationship Matrix											
Semester	**Course code**		**Title of the Course**							**Hours**	**Credits**
1	**23PCA1AE01**		**Ability Enhancement Course:** Programming in Java							**2**	**1**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**2**	**3**	**2**	**1**	**1**	**3**	**3**	**3**	**2**	**3**	**2.3**
CO2	**3**	**3**	**2**	**2**	**1**	**3**	**2**	**3**	**2**	**2**	**2.32**
CO3	**3**	**3**	**3**	**2**	**2**	**3**	**3**	**2**	**2**	**2**	**2.5**
Mean overall Score											**2.4 (High)**

Semester	Course Code	Title of the Course	Hours	Credits
1	23PCA1BC01	Bridge Course	30	2

Course Objectives
To provide the basic Concepts in Information Technology
To provide the concepts of mathematical logic and discrete structures
To provide the mathematical logic for solving problems
To understand the fundamental syntax and concepts of C programming, Control tatements and Looping structures
To write programs using various control structures, strings, arrays and pointers.

UNIT I: Fundamentals of Information Technology

Introduction to Computers – Generation of Computers – Classification of Digital Computer – Anatomy of Digital Computer. CPU and Memory – Secondary Storage Devices – Input Devices – Output Devices. Introduction to Computer Software – Programming Language – Operating Systems - Introduction to Database Management System.

UNIT II: Mathematical Foundations for Computer Science

Mathematical Logic: Statements and Notation – Connectives-Statement. Formulas and Truth Tables – Tautologies – Equivalence of Formulas – Duality Law. Tautological implications – Theory of inference – validity using truth tables – Rules of Inference.

UNIT III: Problem Solving Techniques

Algorithms – Flow charts – Developing algorithms and flowcharts for solving simple problems using sequential, selection and iterative programming Structures.

UNIT IV: Programming In C

Structure of a C program – Data Types – Constants and Variables – Operators and Expressions – Control structures – Looping structures. Arrays – Functions – Built-in-functions – User defined functions – Scope of Variables – Passing Arrays to function – Strings and pointers.

UNIT V: Coding Practices

Simple Programs using Operators – Branching structures – Looping structures – Arrays Strings – Functions – Structures – Union – Pointers.

Teaching Methodology	Lecture-based instruction, Project-based learning, Demonstration

Books for Study

1. Leon, A. & Leon, M. (2009). *Fundamentals of Information Technology* (2nd ed.). Vikas Publishing House Pvt. Ltd.
2. Tremblay, J. P. & Manohar, R. (2008). *Discrete Mathematical Structures with Applications to Computer Science* (1st ed.). McGraw-Hill International Edition.

3. Jaiswal, S. (2009). *Information Technology Today* (4th ed.). Galgotia Publications.
4. Balagurusamy, E. (2016). *Programming in ANSI C* (7th ed.). Tata McGraw Hill Education Private Limited.

Books for Reference

1. Gottfried, B. & Schaum. (2018). *Outline Programming with C* (4th ed.). Tata McGraw Hill Education Private Limited.
2. Kernighan & Ritchie. (1998). *The C Programming Language* (2nd ed.). Prentice Hall.
3. Kanetkar, Y. (2021). *Let Us C* (18th ed.). BPB Publications.

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2CC04	**Core Course - 4:** Programming Smart Devices	4	4

Course Objectives
To learn the fundamental elements of React Native application
To create an elegant UI design using React Native elements
To understand and use various device facilities in the React Native application
To learn the art of communicating with web servers from React Native application
To gain knowledge on deployment of React Native application in Android play store and iOS app store

UNIT I: Learning the basics **(12 Hours)**

Introduction to React - Virtual Document Object Model (DOM) - One-way data flow - Introduction to components - Props and state -Introduction to React Native - The installation of React Native - First application - The anatomy of a React Native application - Debug your application.

UNIT II: UI Design **(12 Hours)**

React Navigation - Flex box - Touchable Highlight - List View - Scroll View - Animations - Image scrolling and swiping.

UNIT III: Device Capabilities **(12 Hours)**

Map View and Geo Location - Async Storage - Native Alert - Web View - Deep linking.

UNIT IV: Communicating with Servers **(12 Hours)**

XML Http Request - Web Socket - Fetch - Getting data from a server - Saving data to a server - Creating APIs with Node/Express - Integrating RN app with Node server.

UNIT V: React Native Application Distribution **(12 Hours)**

The Apple and Google Play distribution systems - Creating a build for iOS or Android - Beta testing.

Teaching Methodology	Live Demonstration of App development, Hands-on Labs, Group Projects and Collaborations, Industry Experts Lecture

Books for Study

1. Paul, A., & Nalwaya, A. (2019). *React Native for Mobile Development* (6th ed.). Tata McGraw Hill.

UNIT I : Chapters 1 and 2
UNIT II : Chapter 4
UNIT III : Chapter 5
UNIT IV : Chapter 6
UNIT V : Chapter 9

Books for Reference

1. Masiello, E., & Friedmann, J. (2017). *Mastering React Native*. Birmingham.
2. Eisenman, B. (2016). *Learning React Native - Building Mobile Applications with JavaScript*. O'Reilly Media.
3. Dabit, N. (2019). *React Native in Action - Developing iOS and Android apps with JavaScript*. Manning Publication & Co.

Websites and eLearning Sources

1. React Native Official Documentation - https://reactnative.dev/
2. React Native Elements - https://reactnativeelements.com/

3. React Native Navigation - https://reactnavigation.org/

Course Outcomes		
CO No.	**CO-Statements**	**Cognitive Levels (K- Level)**
	On successful completion of this course, students will be able to	
CO1	recall the features of React Native in the making of cross-platform Mobile applications.	**K1**
CO2	understand the concepts used in smart devices by creating simple applications using react native.	**K2**
CO3	construct powerful and elegant mobile applications using React components.	**K3**
CO4	test mobile apps that interact with APIs on the server-side.	**K4**
CO5	evaluate the use of features in building mobile applications.	**K5**
CO6	develop applications and deploy them on iOS App Store and Android Play Store.	**K6**

Relationship Matrix											
Semester	**Course Code**			**Title of the Course**					**Hours**		**Credits**
2	**23PCA2CC04**			**Core Course - 4:** Programming Smart Devices					**4**		**4**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Scores of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	3	2	3	2	2	3	2	2	2	3	**2.4**
CO2	2	3	3	2	3	2	3	2	2	3	**2.5**
CO3	3	2	3	2	2	3	2	2	2	2	**2.3**
CO4	3	3	2	2	2	3	3	3	2	3	**2.6**
CO5	2	3	3	2	1	3	3	2	2	3	**2.4**
CO6	2	3	3	2	2	3	3	2	2	3	**2.5**
										Mean Overall Score	**2.45 (High)**

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2CC05	**Core Course - 5:** Software Engineering	4	3

Course Objectives
To understand the fundamental principles and evolving nature of software engineering.
To develop proficiency in requirements engineering and the elicitation of software requirements.
To acquire design concepts and skills, emphasizing architectural design and user interface design.
To learn diverse testing strategies applicable to conventional, object-oriented, web, and mobile applications.
To gain knowledge of project management concepts and the effective application of metrics for process improvement and software quality assurance.

UNIT I : The Nature of Software **(12 Hours)**

The Changing Nature of Software - Software Engineering: Defining the Discipline - The Software Process - Software Engineering Process - Software Development Myths. Process Models: Prescriptive Process Models - Specialized Process Models - The Unified Process - Personal and Team Process Models - Process Technology - Product and Process. Agile Development: Meaning of Agility and Cost of Change - Agile Process - Extreme Programming - Other Agile Process Models - A Tool Set for the Agile Process.

UNIT II: Understanding Requirements **(12 Hours)**

Requirements Engineering - Establishing Groundwork - Eliciting Requirements - Developing Use Cases - Building the analysis Model - Negotiating Requirements - Requirements Monitoring - Validating Requirements - Avoiding common mistakes. Scenario-Based Methods: Requirements Analysis - Scenario-Based Modeling - UML models that supplement the use cases. Class-Based Methods: Identifying Analysis Classes - Specifying Attributes - Defining Operations - Class-Responsibility - Collaborator Modeling - Associations and Dependencies - Analysis Packages.

UNIT III: Design Concepts **(12 Hours)**

The Design Process - Design Concepts - The Design Model. Architectural Design: Software Architecture - Architectural Genres - Architectural Styles - Architectural Considerations - Architectural Decisions - Architectural Design - Assessing Alternative Architectural Design. User Interface Design: The Golden Rules - User Interface Analysis and Design - Interface Analysis - Interface Design Steps - WebApp and Mobile Interface Design - Design Evaluation.

UNIT IV: Software Testing Strategies **(12 Hours)**

A Strategic Approach to Software Testing - Test Strategies for Conventional Software - Test Strategies for Object-Oriented Software - Test Strategies for WebApp - Test Strategies for Mobile App - Validation Testing - System Testing - The Art of Debugging. Testing Conventional Applications: Software Testing Fundamentals - Internal and External Views of Testing - White-Box Testing - Basis Path Testing - Control Structure Testing - Black-Box Testing - Model Based Testing - Testing Documentation and help facilities - Testing for Real Time Systems - Pattern for Software Testing.

UNIT V: Project Management Concepts **(12 Hours)**

The Management Spectrum - People - The Product - The Process - The Project - W5H Principle - Critical Process. Process and Project Metrics: Metric in the Process and Project Domains - Software Measurement - Metrics for Software Quality - Integrating Metrics within the Software Process - Metrics for small Organizations - Establishing a Software Metrics Program.

Teaching Methodology	Lectures and Presentations, Interactive Discussions, Case Studies, Collaborative Learning

Books for Study

1. Pressman, R. S., & Maxim, B. (2019). *Software Engineering*, (8th Ed.). McGraw Hill.

Books for Reference

1. Pressman, R. S. (2019). *Software Engineering,* (9th Ed.). McGraw Hill.
2. Sommerville, I. (2018). *Software Engineering,* (10th Ed.). Pearson India.
3. Fairley, R. (2017). *Software Engineering Concepts*. McGraw Hill.

Websites and e-Learning Sources

1. https://www.tutorialspoint.com/software_engineering/index.html
2. https://www.javatpoint.com/software-engineering

Course Outcomes		
CO No.	**CO-Statements**	**Cognitive Levels (K-level)**
	On successful completion of this course, students will be able to	
CO1	recall the basic concepts of Software Engineering	**K1**
CO2	interpret the necessities to develop the Software	**K2**
CO3	apply the methods and techniques in practical projects	**K3**
CO4	compare the various software development methods and understand the context in which each approach might be applicable in real world concept.	**K4**
CO5	evaluate the effectiveness of an organization's software development practices and suggest improvements	**K5**
CO6	build the tools and techniques for large-scale software systems development	**K6**

Relationship Matrix											
Semester	**Course Code**			**Title of the Course**					**Hours**	**Credits**	
2	**23PCA2CC05**			**Core Course - 5:** Software Engineering					**4**	**3**	
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Scores of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	3	3	3	2	1	3	2	3	2	3	**2.5**
CO2	2	3	3	2	2	2	3	2	1	3	**2.3**
CO3	3	2	3	2	2	3	2	2	2	2	**2.3**
CO4	3	3	2	2	2	3	3	3	2	3	**2.6**
CO5	2	3	3	2	1	3	3	2	2	3	**2.4**
CO6	2	3	3	2	1	3	3	2	2	3	**2.4**
										Mean Overall Score	**2.4 (High)**

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2CC06	**Core Course - 6:** Data Analysis Using Python	4	3

Course Objectives
To understanding of fundamental data types, operations, functions, modules, packages and built-in modules in Python.
To apply Regular Expression Modifiers, creating tables and connecting to databases.
To acquire skills on NumPy and effectively working with NumPy structured arrays.
To import data manipulation skills using Pandas and perform vectorized string operations.
To enhance data visualizing skills using Matplotlib and Seaborn.

UNIT I: Introduction to Python **(12 Hours)**

Features of Python - Data Types and Operations: Numbers-Strings-List-Tuple-Set-Dictionary. Functions: Function Definition -Function Calling -Function Arguments-Anonymous Functions. Modules and Packages: Built-in Modules - Creating Modules - import Statement - Locating Modules - Namespaces and Scope - dir() function - reload() function - Packages in Python -Date and Time Modules.

UNIT II: Regular Expressions and Database Programming **(12 Hours)**

match () function - search() function - Search and Replace - Regular Expression Modifiers: Option Flags-Regular Expression Patterns - findall() method-compile() method. Database Programming: Connecting to a Database-Creating Tables-Insert, Update, Delete and Read Operation- Disconnecting from a Database.

UNIT III: Numpy **(12 Hours)**

Introduction to Numpy-Basics of NumPy Array-Computation on NumPy Array - Aggregations - Broadcasting - Comparisons, Masks and Boolean Logic- Sorting Arrays - NumPy Structured Array.

UNIT IV: Pandas **(12 Hours)**

Data Manipulation with Pandas: Introducing Panda Objects - Data Indexing and Selection -Operating Data on Pandas - Handling Missing Data - Hierarchical Indexing -Combining Data Sets- Vectorized String Operations- Working with Time Series.

UNIT V: Matplotlib **(12 Hours)**

Visualization with Matplotlib: Simple Line Plots-Simple Scatter Plots-Density and Contour Plots-Histograms, Binnings and Density-Customizing Plot Legends -Customising Colorbars-Multiple Subplots-Textand Annotation-Three Dimension Plottingin Matplotlib-Geographic Data with Base Map-Visualization with Seaborn.

Teaching Methodology	Lecture-based instruction, Demonstration, Group Discussion, Peer Learning, Problems solving, and Project-based learning,

Books for Study

1. Jose, J., & Sojan Lal, P. (2016). *Introduction to Computing and Problem Solving with PYTHON*. Khanna Book Publishing Co. (P) Ltd.
   UNIT I: Chapter 3 (3.1, 3.2), Chapter 4 (4.1-4.6), Chapter 6 (6.1-6.4), Chapter 7 (7.1-7.9)
   UNIT II: Chapter 11 (11.1-11.5, 11.9, 11), Chapter 12 (12.1-12.6, 12.8)
2. Vander Plas, J. (2016). *Python Data Science Handbook: Essential Tools for Working with Data* (1st Ed.). O'Reilly Media.
   UNIT III: Chapter 2 UNIT IV: Chapter 3 UNIT V: Chapter 4

Books for Reference

1. Chun, W. J. (2006). *Core Python Programming,* (2nd Ed.). Prentice Hall Publication.
2. Budd, T. A. (2011*). Exploring Python*. Tata McGraw Hill.
3. Boschetti, A., & Massaron, L. (2018). *Python Data Science Essentials,* (3rd Ed.). Pack Publishing.

Websites and eLearning Sources

1. https://realpython.com/
2. https://towardsdatascience.com/
3. https://jupyter.org/
4. https://pandas.pydata.org/pandas-docs/stable/

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K-level)**
CO1	acquire knowledge about various programming constructs and libraries like Numpy, Pandas and Matplotlib used for data analysis in Python.	**K1**
CO2	explain the basic concepts of object-oriented & procedural programming and concepts used in various data analysis libraries like Numpy, Pandas and Matplotlib available in Python.	**K2**
CO3	apply core Python concepts to write simple programs and various libraries like Numpy, Pandas and Matplotlib used in Python for performing data analysis.	**K3**
CO4	discover how to implement core python concepts in various domains and data analysis using various libraries like Numpy, Pandas and Matplotlib.	**K4**
CO5	assess simple Python applications to perform data analysis using various libraries.	**K5**
CO6	develop Python applications and perform data analysis using various libraries like Numpy, Pandas and Matplotlib.	**K6**

Relationship Matrix											
Semester	**Course Code**			**Title of the Course**						**Hours**	**Credits**
2	**23PCA2CC06**			**Core Course - 6:** Data Analysis Using Python						**4**	**3**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Scores of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	3	3	2	2	1	3	3	2	2	2	**2.3**
CO2	3	3	2	2	1	3	3	3	2	3	**2.5**
CO3	3	3	2	2	1	3	3	2	2	3	**2.4**
CO4	3	3	3	2	1	3	3	3	2	3	**2.6**
CO5	3	3	3	2	1	3	3	3	2	3	**2.6**
CO6	3	3	3	2	1	3	3	3	2	3	**2.6**
Mean Overall Score											**2.5 (High)**

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2CP03	**Core Practical - 3:** Programming Smart Devices	3	3

List of Exercises

1. Creating a simple React Native application
2. Responsive UI design with Flex
3. UI design with components
4. Use of front-end tools and event handling
5. Implementation of touch
6. Navigation among screens
7. Swipe feature implementation
8. Creation of API using Node/Express
9. Accessing APIs from React Native application
10. Implementation of interaction with Firebase
11. Implementation of interaction with RDBMS (MySQL)

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2CP04	**Core Practical - 4:** Python Programming	2	2

List of Exercises

Basic Python Programs

1. Flow controls, Functions and String Manipulation
2. Operations on Tuples and Lists
3. Operations on Sets and Dictionary
4. Regular Expressions
5. Database Operations

Data Analysis - NumPy

6. NumPy Arrays,
7. Sorting and Searching on Arrays

Data Analysis - Pandas

8. Data Series
9. Data Frame
10. Combining and Merging Data Sets
11. Handling Missing Values, Filter, Grouping and Aggregation

Visualization - MatplotLib & Seaborn

12. MatPlotLib - Line Chart, Scatter Plot, Histogram
13. Seaborn - Boxplot, HeatMap

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2SP01	**Self-paced Learning:** XML	-	2

Course Objectives
To understand the concepts of XML Markup.
To apply XML Validation Techniques.
To explore XML Transformation with XSLT.
To apply XML in Real-world Scenarios.
To Integrate XML with Programming Languages.

UNIT I: Introducing XML

An Eagle's Eye View of XML - XML Document - Structuring Data - Attributes, Empty - Element Tags and XSL - Well-formedness.

UNIT II: Document Type Definition

Validity - Element Declarations - Attribute Declarations - Entity Declarations -- Namespaces.

UNIT III: Style Languages

CSS Style Sheets - CSS Layouts - CSS Text Styles - XSL Transformations - XSL Formatting Objects.

UNIT IV: Supplemental Technologies

XLinks - Xpointers - Xinclude - Schemas.

UNIT V: XML Applications

Chemical Markup Language - Mathematical Markup Language - RSS Classic literature - Synchronized Multimedia Integration Language - Open Software Description -Scalable Vector Graphics- Music XML - Voice XML.

Teaching Methodology	Online, PPT

Books for Study

1. Harold, E. R. (2004). *XML Bible,* (3rd Ed.). John Wiley & Sons Inc.

Books for Reference

1. Fawcett, J., Quin, L. R. E., & Ayers, D. (2012). *Beginning XML* (5th Ed.). John Wiley & Sons Inc.
2. Powell, T. A. (2010). *The Complete Reference XML* (5th Ed.). The McGraw-Hill Companies.
3. Holzner, S. (2004). *XML in 21 Days* (3rd Ed.). Sams Publishing.

Websites and eLearning Sources

1. https://www.w3.org/TR/xml/
2. https://developer.mozilla.org/en-US/docs/Web/XML/XML_introduction

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2ES03A	**Elective - 3:** Internet of Things	5	4

Course Objectives
To grasp the core concepts and technologies behind the Internet of Things (IoT) and explore practical applications.
To learn the principles for creating connected devices, focusing on communication technologies and gateway-level data management.
To gain proficiency in communication protocols and web connectivity solutions for connected devices.
To Analyze IP addressing, MAC layer, and application protocols for efficient data handling and processing.
To learn to collect, store, and process data using cloud platforms in IoT applications, leveraging cloud computing paradigms and service models.

UNIT I: Internet of Things: An Overview **(15 Hours)**

Internet of Things - IoT Conceptual Framework - IoT Architectural View - Technology Behind IoT - Sources of IoT - M2M Communication - Examples of IoT.

UNIT II: Design Principles for Connected Devices **(15 Hours)**

Introduction - IoT/M2M Systems Layers and Design Standardisation - Communication Technologies - Data Enrichment, Data Consolidation and Device Management at Gateway.

UNIT III: Design Principles for Web Connectivity **(15 Hours)**

Introduction - Web Communication Protocols for Connected Devices - Message Communication Protocols for Connected Devices - Web Connectivity for Connected Devices Network using Gateway, SOAP, REST, HTTP RESTful and WebSockets.

UNIT IV: Internet Connectivity Principles **(15 Hours)**

Introduction - Internet Connectivity - Internet Based Communication - IP Addressing in IoT - Media Access Control - Application Layer Protocols: HTTP, HTTPS, FTP, Telnet and Others. Data Acquiring, Organising, Processing and Analytics: Data Acquiring and Storage - Organising the Data Analytics.

UNIT V: Data Collection, Storage and Computing using Cloud Platforms **(15 Hours)**

Cloud Computing Paradigm for Data Collection, Storage and Computing - Everything as a Service and Cloud Service Models. Sensor and Wireless Sensor Networks: Sensor Technology - Participatory Sensing, Industry IoT and Automative IoT - Actuator - Sensor Data Communication Protocols - Radio Frequency Identification Technology - Wireless Sensor Networks Technology.

Teaching Methodology	Tutorials, Demonstration & IoT Simulations

Books for Study

1. Kamal, R. (2017). *Internet of Things: Architecture and Design Principles,* (1st Ed.). McGraw Hill Education (India) Private Limited.
   UNIT I: Chapter 1
   UNIT II: Chapter 2
   UNIT III: Chapter 3
   UNIT IV: Chapters 4, 5
   UNIT V: Chapters 6, 7

Books for Reference

1. Vasudevan, S. K., Nagarajan, A. S., & Sundaran, R. M. D. (2020). *Internet of Things* (2nd Ed.). Wiley Publication.
2. Hanes, D., Salgueiro, G., Grossetete, P., Barton, R., & Henry, J. (2017). *IoT fundamentals: Networking technologies, protocols, and use cases for the Internet of Things.* Cisco Press.
3. Hassan, Q. F. (2018). *Internet of Things A to Z: Technologies and Applications*. Wiley Publication. IEEE Press

Websites and eLearning Sources

1. https://www.shiksha.com/online-courses/industrial-internet-of-things-iiot-course-courl405
2. https://www.tinkercad.com/
3. https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT
4. https://www.oracle.com/in/internet-of-things/what-is-iot/
5. https://www.ibm.com/topics/internet-of-things

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K-Level)**
CO1	recognize key IoT concepts and terminologies to establish a foundational understanding of the Internet of Things.	**K1**
CO2	apply design principles to create connected devices, demonstrating the practical application of IoT/M2M system layers and communication technologies.	**K2**
CO3	implement web connectivity solutions for connected devices, utilizing a range of communication protocols and demonstrating proficiency in IoT network design.	**K3**
CO4	analyze and evaluate internet connectivity principles, including IP addressing, MAC layer, and application protocols, demonstrating critical thinking skills.	**K4**
CO5	synthesize data acquisition, organization, and processing techniques for IoT applications, showcasing advanced problem-solving abilities.	**K5**
CO6	evaluate the integration of cloud computing paradigms for efficient data management in IoT applications, demonstrating a comprehensive understanding and the ability to make informed decisions.	**K6**

Relationship Matrix											
Semester	**Course Code**			**Title of the Course**						**Hours**	**Credits**
2	**23PCA2ES03A**			**Elective - 3:** Internet of Things						**5**	**4**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Scores of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	2	3	3	2	1	3	2	3	2	2	**2.3**
CO2	3	3	3	3	2	3	3	3	2	2	**2.7**
CO3	2	3	2	2	1	3	3	2	2	1	**2.1**
CO4	3	3	3	2	2	3	3	3	3	2	**2.7**
CO5	3	3	3	3	2	3	3	3	3	2	**2.8**
CO6	3	3	2	2	2	2	3	3	2	3	**2.5**
										Mean Overall Score	**2.5 (High)**

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2ES03B	**Elective - 3:** Cloud Computing	5	4

Course Objectives
To comprehend cloud computing fundamentals like virtualization, resource pooling and on-demand self-service.
To apply public, private, and hybrid cloud architectures to solve real-world problems and design efficient solutions.
To gain practical skills in deploying, managing, and optimizing cloud applications and infrastructure.
To develop a deep understanding of cost management in cloud environments and design them for cloud services.
To explore advanced cloud topics including security, scalability, virtual networks, and compliance for comprehensive cloud solution analysis.

UNIT I: Fundamentals of Cloud Computing **(15 Hours)**

Origin and Influences - Business Drivers - Technology Innovations - Basic Concepts - Scaling - Risks and Challenges - Roles and Boundaries - Cloud Characteristics - Cloud Delivery Models - Cloud Deployment Models - Internet Architecture - Data Center Technology - Virtualization Technology - Web Technology.

UNIT II: Cloud Computing Mechanisms **(15 Hours)**

Cloud Infrastructure Mechanisms - Logical Network Perimeter - Cloud Storage Device - Cloud Usage Monitor - Specialized Cloud Mechanisms - Cloud Management Mechanisms - Fundamental Cloud Security - Cloud Security Mechanisms.

UNIT III: Cloud Computing Architecture **(15 Hours)**

Fundamental Cloud Architectures - Advanced Cloud Architectures - Specialized Cloud Architectures.

UNIT IV: Working with Clouds **(15 Hours)**

Delivery Model Considerations - Consumer Perspective - Cost Metrics and Pricing Models - Cost Management Considerations - Service Quality Metrics.

UNIT V: Virtual Networks and other aspects of Cloud **(15 Hours)**

Virtual Machines - Approaches to Virtualization - Properties of Full Virtualization - Organization of VM Systems - Levels of Trust - Virtual I/O Devices - VM Migration - Live Migration - Running Virtual Machines in an Application - Hosted Hypervisor. Edge Computing and IIoT - Latency of Cloud - Edge to Fog Hierarchy - Communication for IIoT - Decentralization.

Teaching Methodology	Videos, PPT, Case Studies, Demonstration, and Hands on sessions

Books for Study

1. Erl, T., Mahmood, Z., & Puttini, R. (2013). *Cloud Computing Concepts, Technology & Architecture*. Prentice Hall. (UNIT I, II, III and IV)
2. Comer, D. E. (2021). *The Cloud Computing Book: The Future of Computing Explained,* (1st Ed.). CRC Press. (Unit V).

Books for Reference

1. Buyya, R., Broberg, J., & Broberg, J. (2011). *Cloud Computing: Principles and Paradigms*. Wileys.
2. Baron, S. (2020). *AWS: The Complete Beginner's Guide to Mastering Amazon Web Services*. Independently Published.
3. Vergadia, P. (2022). *Visualizing Google Cloud: Illustrated References for Cloud Engineers &*

Architects: 101 Illustrated References for Cloud Engineers and Architects, (1st Ed.). John Wiley & Sons Inc.

Websites and eLearning Sources

1. https://docs.aws.amazon.com/
2. https://cloud.google.com/docs
3. https://learn.microsoft.com/bs-latn-ba/azure/cloud-services/
4. https://www.vmware.com/in/products/workstation-pro.html

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K - Level)**
CO1	recall fundamental cloud computing concepts, such as its origins and key drivers.	**K1**
CO2	explain the core mechanisms of cloud computing, including storage, security, and network components.	**K2**
CO3	design practical cloud solutions based on specific requirements, leveraging knowledge of architecture and mechanisms.	**K3**
CO4	evaluate the risks and challenges associated with cloud adoption and propose mitigation strategies.	**K4**
CO5	critically evaluate the security mechanisms and best practices for securing cloud environments.	**K5**
CO6	create comprehensive cloud strategies that incorporate advanced technologies and address complex challenges.	**K6**

Relationship Matrix											
Semester	**Course Code**			**Title of the Course**					**Hours**		**Credits**
2	**23PCA2ES03B**			**Elective - 3:** Cloud Computing					**5**		**4**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Scores of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	2	3	2	3	3	3	3	3	2	3	**2.7**
CO2	2	3	2	2	3	2	3	2	2	2	**2.3**
CO3	3	3	2	2	3	2	2	3	2	2	**2.3**
CO4	2	2	2	3	2	3	2	2	2	3	**2.3**
CO5	3	3	2	2	1	1	2	3	2	3	**2.2**
CO6	3	3	2	2	2	2	2	2	2	2	**2.2**
										Mean Overall Score	**2.3 (High)**

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PSS2SE01	**Skill Enhancement Course:** Soft Skills	4	3

Course Objectives
To provide a focused training on soft skills for students in colleges for better job prospects
To communicate effectively and professionally
To help the students take active part in group dynamics
To familiarize students with numeracy skills for quick problem solving
To make the students appraise themselves and assess others

Unit I: Effective Communication & Professional Communication (12 Hours)

Definition of communication, Barriers of Communication, Non-verbal Communication; Effective Communication - Conversation Techniques, Good manners and Etiquettes; Speech Preparations & Presentations; Professional Communication.

Unit II: Resume Writing & Interview Skills (12 Hours)

Resume Writing: What is a résumé? Types of résumés, - Chronological, Functional and Mixed Resume, Purpose and Structure of a Resume, Model Resume.

Interview Skills: Types of Interviews, Preparation for an interview, Attire, Body Language, Common interview questions, Mock interviews & Practicum

Unit III: Group Discussion & Personal effectiveness (12 Hours)

Basics of Group Discussion, Parameters of GD, Topics for Practice, Mock GD & Practicum & Team Building.

Personal Effectiveness: Self Discovery; Goal Setting with questionnaires & Exercises

Unit IV: Numerical Ability (12 Hours)

Introducing concepts Average, Percentage; Profit and Loss, Simple Interest, Compound Interest; Time and Work, Pipes and Cisterns.

Unit V: Test of Reasoning (12 Hours)

Introducing Verbal Reasoning: Series Completion, Analogy; Data Sufficiency, Assertion and Reasoning; and Logical Deduction. Non-Verbal Reasoning: Series; and Classification

Teaching Methodology	Chalk and talk, Lectures, Demonstrations, PPT.

Book for study

1. Melchias G., Balaiah, J. & Joy, J. L. (Eds). (2018). *Winner in the Making: A Primer on soft Skills.* Trichy, India: St. Joseph's College.

Books for References

1. Aggarwal, R. S. (2010). *A Modern Approach to Verbal and Non-Verbal Reasoning.* S. Chand.
2. Covey, S. (2004). *7 Habits of Highly effective people.* Free Press.
3. Gerard, E. (1994). *The Skilled Helper* (5th Ed.). Brooks/Cole.
4. Khera, S. (2003). *You Can Win.* Macmillan Books.
5. Murphy, R. (1998). *Essential English Grammar*, (2nd Ed.). Cambridge University Press.
6. Sankaran, K., & Kumar, M. (2010). *Group Discussion and Public Speaking* (5th Ed.). M.I. Publications.
7. Trishna, K. S. (2012). *How to do well in GDs & Interview*s? (3rd Ed.). Pearson Education.
8. Yate, M. (2005). *Hiring the Best: A Manager's Guide to Effective Interviewing and Recruiting*

Course Outcomes		
CO No.	CO-Statements On successful completion of this course, students will be able to	Cognitive Levels (K - Level)
CO1	recall various soft skill sets	K1
CO2	understand personal effectiveness in any managerial positions	K2
CO3	apply verbal and non-verbal reasoning skills to solve problems	K3
CO4	differentiate problems at work and home; and design solutions to maintain work-life balance	K4
CO5	assess growth and sustainability and infuse creativity in employment that increases professional productivity	K5
CO6	construct plans and strategies to work for better human society	K6

Relationship Matrix											
Semester	Course Code			Title of the Course				Hours			Credits
2	23PSS2SE01			Skill Enhancement Course: Soft Skills				4			3
Course Outcomes	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Scores of COs
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	3	3	3	3	2	3	2	3	2	3	2.7
CO2	3	3	3	2	3	3	3	3	3	3	2.9
CO3	3	2	2	3	3	3	3	3	3	3	2.8
CO4	3	3	2	2	3	3	3	3	3	3	2.8
CO5	3	3	3	2	2	3	3	3	3	3	2.8
CO6	3	3	3	2	2	3	3	3	3	3	2.8
Mean Overall Score											2.8 (High)

Semester	Course Code	Title of the Course	Hours/Week	Credits
2	23PCA2EG01	**Generic Elective - 1 (WS):** Applied Statistics Using R	4	3

Course Objectives
To provide an overview of R Language, Data categorization and basic programming constructs in R.
To understand various R data structures like vector, matrix and list.
To explore Data Frames and Tables in R and brief concepts about Graphics in R.
To introduce the concepts on descriptive statistics and probability.
To acquaint knowledge the R commands related to Correlation & Regression, Statistical inference and Variance of Analysis.

UNIT I: Introduction To R Programming **(12 Hours)**

Overview of R - Installation of R - Loading R Packages - R Basic Syntax - Data Types and Objects - Variables - Constants - Comments - Debuting in R. Data Definition And Categorisation: Overview of Data - Sources of Data - Big Data - Data Categorisation - Data Cube. Control Statements and Functions: if Statement - for statement - while loop - repeat and break Statements - next Statement - switch Statement - Functions.

UNIT II: Vectors, Matrix and List **(12 Hours)**

Overview of Vector - Creating a Vector - Accessing Elements of a Vector - Vector Manipulation and Vector Arithmetic - Deleting a Vector - Vector Element Sorting. Matrices: Creating a Matrix -Matrix Subsetting - Matrix Operations - Combining Matrices - Special Matrices - Eigen Vectors and Eigen Values - Arrays LISTS: Introduction to Lists - Creating a List - General List Operations - Accessing and Manipulating Elements of a List - Merging Lists - Applying Functions to a List - Sorting and Searching.

UNIT III: Data Frames, Factors and Graphics in R **(12 Hours)**

Introduction to Data Frames - Creating a Data Frame - General Operations on Data Frames - Extending a Data Frame - Applying Functions to Data Frame. Factors and Tables: Introduction to Factors - Creating a Factor - Factor Levels - Summarising a Factor - Ordered Factors - Converting Factors - Common Functions used with Factors - Introduction to Tables and Creating Tables - Table Related Functions - Cross Tabulation. Graphics in R: Creating Graphs - Histograms - Bar Plot - Line Chart - Pie Chart - Box Plot _ Scatter Plot - Saving Graphs to a File

UNIT IV: Descriptive Statistics and Probability Using R **(12 Hours)**

Introduction to Statistical Analysis in R - Measures of Central Tendency or Location - Measures of Dispersion - Measures of Shape. Probability: Introduction to Probability - Probability and Statistics - Random Variables - Probability Distribution.

UNIT V: Correlation, Regression and ANOVA **(12 Hours)**

Correlation Analysis - Regression Analysis. Statistical Inference: Introduction to Statistical Inference - Hypothesis Testing. Analysis of Variance: Introduction to Analysis of Variance - Implementing Analysis of Variance - ANOVA in R.

Teaching Methodology	Videos, PPT, Demonstration, and Hands on sessions

Book for Study

1. Arora, S., & Malik, L. (2020). R *Programming for Beginners*. Universities Press India (P) Ltd.
   - **UNIT I** - Chapters 1, 2 & 4
   - **UNIT II** - Chapters 6, 7 & 8
   - **UNIT III** - Chapters 9, 10 & 14

UNIT IV - Chapters 17 & 18
UNIT V - Chapters 20, 21 & 22

Books for Reference

1. Matloff, N. (2011). *The Art of R Programming: A Tour of Statistical Software Design*. Starch Press.
2. Gardener, M. (2013). *Beginning R - The Statistical Programming Language*. Wiley.
3. Verma, A. K. (2017). *R Programming*. Cengage Learning.

Websites and eLearning Sources

1. https://datatechs.in/courses/r-language/
2. https://www.atnyla.com/syllabus/r-programming-language/7
3. https://www.w3schools.com/r/
4. https://www.geeksforgeeks.org/r-tutorial/
5. https://www.javatpoint.com/r-tutorial

Course Outcomes		
CO No.	**CO-Statements** On successful completion of this course, students will be able to	**Cognitive Levels (K- Level)**
CO1	recall the basic concepts used in R Programming structure	**K1**
CO2	compare different data structures in R like Vectors, Matrices and Lists.	**K2**
CO3	apply fundamental statistical techniques in R tool.	**K3**
CO4	determine the use of R commands in an Integrated Application Development.	**K4**
CO5	compare R tool with other statistical tools in market.	**K5**
CO6	develop basic R scripts using real time datasets to solve problems, create plots and charts for data representation, and interpret basic statistical results obtained through R.	**K6**

Relationship Matrix											
Semester	**Course Code**		**Title of the Course**							**Hours**	**Credits**
2	**23PCA2EG01**		**Generic Elective - 1 (WS):** Applied Statistics Using R							**4**	**3**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	3	3	2	2	1	3	3	2	2	2	**2.3**
CO2	3	3	2	2	1	3	3	3	2	3	**2.5**
CO3	3	3	2	2	1	3	3	2	2	3	**2.4**
CO4	2	3	3	2	2	3	3	3	2	3	**2.6**
CO5	3	3	3	2	1	3	3	3	2	3	**2.6**
CO6	3	3	3	2	1	3	3	3	2	3	**2.6**
Mean Overall Score											**2.5 (High)**