Supporting Document for 1.1.2

Syllabus Revision carried out in 2019

Program: MCA- Master of Computer Applications
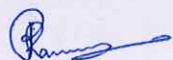
Regulations : SVEC-19

This document details the following:

1. Courses where syllabus has been changed 20% and more.
2. Course-wise revised syllabus with changes highlighted.

Note: *For SVEC-19 revised syllabus, SVEC-16 (previous syllabus) is the reference.*

List of Courses where syllabus has been changed (20% and more)

Course Code	Name of the course	Percentage of Syllabus changed	Page Number in which Details are Highlighted
19MC10131	PC Software Lab	25	2
19MC10132	Programming in C Lab	35	6
19MC20101	Database Management Systems	20	11
19MC20104	Software Engineering	80	17
19MC20131	Database Management Systems Lab	40	22
19MC20132	Data Structures Lab	25	31
19MC30102	Data Warehousing and Data Mining	20	35
19MC30103	Python Programming	100	40
19MC30131	Computer Networks Lab	40	43
19MC30132	Data Warehousing and Data Mining Lab	30	49
19MC30133	Python Programming Lab	100	54
19MC40101	Object Oriented Analysis and Design	20	58
19MC40104	Block Chain Technologies	100	62
19MC40105	Multimedia Application Development	40	64
19MC40106	Information Security	40	68
19MC40109	Artificial Intelligence	100	74
19MC40110	Cyber Security	100	77
19MC40111	Service Oriented Architecture	100	79
Average		56.39	
Total No. of Courses in the Program		28	
No. of Courses where syllabus (more than 20%) has been changed		18	
Percentage of Syllabus changed in the Program		36.25	

DEAN (Academics)

Dean (Academics)
Sree Vidyanikethan Engg. College
Sree Sainath Nagar
TIRUPATI - 517 102, A.P., India.

PRINCIPAL

PRINCIPAL
SREE VIDYANIKETHAN ENGINEERING COLLEGE
(AUTONOMOUS)
Sree Sainath Nagar, A. RANGAMPET
Chittoor (Dist.) - 517 102, A.P., INDIA.

(19MC10131) **PC SOFTWARE LAB**

Int. Marks	Ext. Marks	Total Marks		L	T	.	C
50	50	100		-	-	3	1.5

PRE-REQUISITE: --

COURSE DESCRIPTION:

Peripherals of a computer; Disassembling and Assembling the Personal Computer(PC); Linux file system, File handling utilities and Text processing utilities; Productivity tools including Word, Excel, Power Point, Access and publisher.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Identify and analyze functional parts of PC, functionalities of Operating System and Information Management.

CO2: Apply Troubleshooting techniques for Assembling and Disassembling of PC and create professional word documents, Excel Spreadsheets and power point presentations for effective management of data.

CO3: Work independently and in a team to solve problems with effective communication.

LIST OF EXERCISES:

PC Hardware

1. a) Identification of peripherals of a computer, components in a CPU and its functions.

Construct block diagram of CPU along with the configuration of each peripheral.

b) Demonstration of disassembling and assembling the PC.

2. Demonstrate Operating System, Components of OS, Installation of Microsoft Windows XP Operating System.

3. Implement Basic MS-DOS Internal and External commands.

4. a) Introduction to Linux file system, perform File handling utilities and Text processing Utilities.

b) Implement Shell Script to generate Fibonacci series.

MS Word and LaTeX

5. Implement Mail merge using MS-Word.

6. a) Design a visiting card using MS-Word.

b) Create a project Certificate using LaTeX with the features like Formatting Fonts, Text Effects, Borders and Colors.

MS Excel

7. The ABC Company shows the sales of different products for 5 years.

Create Column Chart, Pie Chart and Bar chart for the following data:

YEAR	PRODUCT-1	PRODUCT-2	PRODUCT-3	PRODUCT-4
2014	1000	800	900	1000
2015	800	80	500	900
2016	1200	190	800	400
2017	400	200	300	1000
2018	1800	400	400	1200

MS Power Point

8. Design a power point presentation on Department of MCA which includes Animations, Design, Sound effects and Images.

MS Access

9. Create Employee and Department tables with the following fields:

Employee table: enumber, ename, salary, deptno.

Department table: deptno, dname, location.

a) Create one-to-many relationship between above two tables.

b) Create a table, form and generate a report with the following fields:

Book_No, Book_Name, Author, Publisher and Price.

10. Design Form and generate Reports for the above employee table and implement the queries as per the given criteria:

a) Retrieve employee details whose salary is above 30,000.

b) Retrieve employee details whose deptno is 20.

11. Mini Project:
Create a Web page with the given features: Home page, About us, Department, Contact page using Microsoft Publisher.

REFERENCE BOOKS:
1. ITL Education Solutions Limited, "*Introduction to Information Technology,*" Pearson Education India, 2nd Edition, 2012.
2. John Walkenbach, Herb Tyson, Michael R. Groh, Faithe Wempen, Lisa A. Bucki, "*Microsoft Office 2010 Bible,*" John Wiley and Sons, 2010.
3. Peter Norton, "*Introduction to Computers*," Tata McGraw-Hill, New Delhi, 7th Edition, 2012.
4. Vikas Gupta, "*Comdex Information Technology Course Tool Kit,*" WILEY Dreamtech, New Delhi, 10th Edition, 2009.
5. Sumitabha Das, "*UNIX Concepts and Applications*," Tata McGraw Hill, 4th Edition, 2017.

ADDITIONAL LEARNING RESOURCES:
1. https://nptel.ac.in/courses/106106144/2
2. https://support.office.com/
3. https://www.latex-project.org/
4. https://www.pcworld.com/

16MC10131: IT LAB

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
50	50	100		-	-	3	2

PREREQUISITES: --

COURSE DESCRIPTION:

Peripherals of a computer and disassembling & assembling the PC; Linux file system and File handling utilities & Text processing utilities; Productivity tools including Word, Excel, Power Point, Access, publisher.

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

CO1. Acquire skills in:

- Identification of Functional parts of PC
- Operating Systems

CO2. Identify the appropriate features to design documents, excel spread sheets and power point presentations.

CO3. Design documents, excel spread sheets, power point presentations, Access database and personal websites effectively.

CO4. Update knowledge and skills in PC maintenance and usage of latest Operating Systems and MS-Office.

CO5. Practice of ethical code of conduct in the usage of computer hardware and software.

CO6. Engage in life-long learning and attitude to work in teams.

CO7. Work together to customize the existing tools.

LIST OF EXERCISES

PC Hardware

1. Identify the peripherals of a computer, components in a CPU and its functions. Block diagram of CPU along with the configuration of each peripheral.

2. Demonstrating disassembling and assembling the PC back to working condition.

3. Introduction to Operating Systems, important of Operating System, components of OS, Installation of Microsoft Windows-XP Operating Systems.

4. Basic MS-DOS commands – Internal and External Commands.

5. Introduction to Linux file system, perform File handling utilities and Text processing utilities.

6. Introduction to Linux - vi editor and Shell Script

7. a) Write a shell Script to generate Fibonacci series.

   b) Write a Shell Script to find factorial of a given number.

MS-Office

MS Word

8. a) Design a visiting card in MS-Word (2''x3.5'').

   b) Perform Mail merge in MS-Word.

MS Excel

9. a) Create a spreadsheet for generating student mark list.

   b) Create a spreadsheet for generating all charts

   c) Import external data to Ms-excel, perform sorting and filter operations on that data.

MS Power Point

10. a) Create text and images with effects.

    b) Prepare a power point presentation on department of MCA which includes Animations, design, sound effects and images.

MS Access

11. Create Access database which consists of at least 3 tables

12. Perform Queries, form design and Reports on above tables.

MS Publisher

13. Create a website using the features: Home page, About us, Department, Contact page etc.

REFERENCE BOOKS:

1. ITL Education, *"Introduction to Information Technology,"* Pearson, 2nd Edition, 2005.

2. John Walken bach, *"Microsoft Office 2010 Bible,"* Wiley India Pvt. Ltd, 2010.

3. Peter Norton, *"Introduction to Computers,"* Tata McGraw-Hill, 7th edition, New Delhi 2012.

4. Vikas Gupta, *"Comdex Information Technology Course Tool Kit,"* WILEY Dreamtech, 2nd edition, New Delhi 2006.

5. Sumitabha Das, *"UNIX Concepts and Applications,"* 4th Edition, TMH, 2008.

MCA I - SEMESTER

(19MC10132) **PROGRAMMING IN C LAB**

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
50	50	100		-	-	3	1.5

PRE-REQUISITES:--

COURSE DESCRIPTION:

Program Design and Problem Solving using the C Programming Language; Control Structures; Functions; Arrays; Strings; Pointers; File I/O and the usage of Preprocessor Directives.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Analyze and implement algorithms using C language Tokens, Input/output formatting styles and Programming components to solve problems.

CO2: Design and develop programs using Control Structures, Loops, Functions, Parameter Passing Mechanism, Command Line Arguments, File Handling to solve complex problems.

CO3: Work independently and in a team to solve problems with effective Communication.

LIST OF EXERCISES:

1. a) Design a flow chart and algorithm to add two numbers and find the largest number among two numbers.
   b) Design a flow chart and algorithm for pinting prime numbers for a given range of numbers.

2. a) Write a C Program to find the sum of individual digits of a positive integer.
   b) Write a C Program to find the roots of a quadratic equation.

3. Write a C language programs to implement non-recursive and recursive functions for the following task:
   a) Calculating Factorial of a given number
   b) GCD of a given numbers

4. a) Write a C Program to Add, Subtract and Multiply Two Matrices Using Functions (Passing arrays as arguments to the function)
   b) Write a C program to determine whether a given string is a palindrome or not.

5. a) Write a C program to Insert a Substring into a given main String at a given Position.
   b) Write a C program to delete 'n' characters from a given position in a given string.

6. a) Write a C program to generate Electricity Bill for different categories of users based on different slabs in each category. (Using Nested if else Statement).
   b) Write a c program to evaluate the following expression using loops:
      $1+x^2/2!+x^4/4!+...$upto 5 terms.

7. Write a menu driven program to read list of student names and perform the following operations using array of character pointers:
    a) To insert a student name
    b) To delete a student name

8. a) Write a program to read in an array of names and sort them.
Note: Use functions and pointers.

   b) Write a program to read and display values of an integer array.
Note: Allocate space dynamically for the array using the malloc ().

9. a) Write a C program to copy contents of one file to another file.
   b) Write a C program to reverse the first 'n' characters in a file.
Note: The file name and n are specified on the command line.

10. a) Write a C program to display the contents of a file.
    b) Write a C program to merge two files into a third file
 Note: The contents of the first file followed by those of the second are put in the third file).

Mini Project-1:

Create a list of Employees with the following fields and calculate minimum and maximum salaries of an employee: Emp_Id, Name: First name, Middle name, Last name, Address: Area, City, State, Age, Salary, Designation.

Mini Project-2: Estimation of Tax for an Employee, given the following conditions:
    If income is less than 1,50,000 then no tax.
    If taxable income is in the range 1,50,001 – 3,00,000 then charge 10% tax.
    If taxable income is in the range 3,00,001 – 5,00,000 then charge 20% tax.
    If taxable income is above 5,00,001 then charge 30% tax.

REFERENCE BOOKS:

1. Byron Gottfried, "*Programming in C*," SchaumOutline Series, Tata MC Grawhill, 3rd Edition, 2017.
2. M.T. Somashekara, "*Problem Solving with C,*" PHI Learning Private Limited, New Delhi, 2009.

MCA I-SEMESTER

16MC10132: PROGRAMMING IN C LAB

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
50	50	100		-	-	3	2

PREREQUISITES: A Course on "Programming in C".

COURSE DESCRIPTION:

Program design and problem solving using the C programming language; Programming topics include control structures, functions, arrays, Strings, pointers, and file I/O and the usage of the preprocessor.

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

CO1. Solve problems using knowledge in

- C Tokens
- Input/output Formatting styles
- Control statements
- Data types

CO2. Design and develop the solutions using the techniques-parameter passing mechanism, command line arguments and handling files for real world problems making use of analysis of algorithms and verification.

CO3. Demonstrate the concepts of C as modular programming language which includes functions, pointers and structures to solve real world complex problems.

CO4. Engage lifelong learning and develop programming competency.

LIST OF EXERCISES

1. a) Write an Algorithm and flow chart to read the name and print the name

b) Write an Algorithm and flow chart to add two numbers.

c) Write an Algorithm and a flow chart to calculate area of square.

d) Write Algorithm and flow chart to find the largest of two numbers.

2. a) Write a C Program to find the sum of individual digits of a positive integer.

b) Write a C Program to find the roots of a quadratic equation.

c) Write a C program to read in a three digit number produce following output

(assuming that the input is 347)

3 hundreds

4 tens

7 units

    d) Write a program to generate Fibonacci series.

3. Write a C non recursive and recursive function for the following task

    a) Calculating Factorial
    b) Swapping the values of two variable
    c) Minimum/maximum value from the given input
    d) Nth Fibonacci number
    e) GCD of a Given Number

4. a) Write a C Program to Add, Subtract and Multiply Two Matrices Using Functions (Passing arrays as arguments to the function)

    b) Write a C program to determine if the given string is a palindrome or not

5. a) Write a C Program to Insert a Substring into a Given Main String from a given Position.

    b) Write a C Program to Delete n Characters from a Given Position in a Give String.

6. a) Write a program to swap two numbers using pointers.

    b) Write a program to find sum of given array using pointers.

7. Write a C program that uses functions to perform the following operations:

    a) Reading a complex number
    b) Writing a complex number
    c) Addition of two complex numbers
    d) Multiplication of two complex numbers

8. a) Write a C program for Electricity Bill Tacking different categories of users, different slabs in each category. (Using Nested If Else Statement)

    b) Write a c program to evaluate the following using loops

        i)1+x2/2!+x4/4!+...up to 5 terms

        ii) x+x3/3!+x5/5!+...up to 5 terms

9. a) Write a c program to check whether the given number is

        i) prime or not
        ii) perfect or abundant or deficient

b) Write a c program to find the mean, mode, median, and variance of list of values by using one dimensional array.

10. a) Write a menu driven program to read a list of numbers and perform the following operations

    i) print the list

    ii) delete duplicates from the list

    iii) reverse the list.

  b) Write a c program that consists of recursive functions to find

    i) factorial of a given number

    ii) print the Pascal triangle using binomial theorem.

11. Write a menu driven program to read list of student names and perform the following operations using array of character pointers. a) to insert a student name

b) to delete a name

c) to print the names.

12. a) Write a C program which copies one file to another.

b) Write a C program to reverse the first n characters in a file.

(Note: The file name and n are specified on the command line.)

13. a) Write a C program to display the contents of a file.

b) Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file)

REFERENCE BOOKS:

1. BS Gottrifried, A.Mittal, *"Programming in C – A practical approach,"* PHI, Tata MC Grawhill, 2008.

2. M.T. Somashekara, *"Problem Solving with C",* PHI Learning Private Limited: New Delhi, 2012.

(19MC20101) DATABASE MANAGEMENT SYSTEMS (Theory)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PRE-REQUISITES:

A course on Operating Systems.

COURSE DESCRIPTION:

Databases; Database Architecture and Database Design; Concepts of Relational Database and its Design; Representation of ER Diagram to Relational model; SQL queries; Normal forms; Recovery and Concurrency Control mechanism; Storage and Indexing mechanism.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Identify and analyze the problems arise due to redundancy, Data normalization techniques, Transaction processing approaches, Concurrency control and Recovery mechanisms for an optimal database application system.

CO2: Select and apply integrity constraints over relations, relational models, storage and indexing, hashing techniques for the construction of relational database systems.

DETAILED SYLLABUS:

UNIT I- INTRODUCTION TO DATABASES, DATABASE ARCHITECTURE AND DATABASE DESIGN (11 Periods)

Databases and Database Users: Concepts, Characteristics of the Database approach, Actors on the Scene, Workers behind the scene, Advantages of the using the DBMS Approach.

Database System Concepts and Architecture: Data Models, Schemas and Instances, Three Schema architecture and Data Independence, Database Languages and Interfaces, Database system environment, Centralized and Client/Server Architectures for DBMS, Classification of Database Management Systems.

Database Design: Database design and ER-diagrams, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER-model, Conceptual Design with the ER-Model.

UNIT II - RELATIONAL MODEL AND BASIC SQL (10 Periods)

Relational Model: Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design: ER to Relational.

Case Study: ER-diagram for banking enterprise and university database.

Basic SQL: SQL data definition and data types, CREATE table command in SQL, attribute data types in SQL, Specifying constraints in SQL: specifying attribute constraints and attribute defaults, specifying key and Referential integrity constraints, specifying CHECK constraints on tuples.

UNIT III - SQL, SCHEMA REFINEMENT AND NORMAL FORMS   (12 Periods)

SQL: Form of Basic SQL Query, Examples of queries in SQL: Insert, Delete and Update statements; Nested Queries, Correlated Nested Queries, Set Comparison Operators, Aggregate Operators, NULL values, Comparison using Null values, Logical connectives: AND, OR , NOT; Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Views, Destroying /altering Tables and Views, Triggers and Active Databases.

Schema Refinement and Normal Forms: Schema Refinement: Problems Caused by redundancy, Decompositions; Problem related to decomposition, Functional Dependencies, Normal Forms: FIRST, SECOND, THIRD Normal forms, BCNF; Properties of Decompositions: Loss less, join Decomposition, Dependency preserving Decomposition.

UNIT IV - OVERVIEW OF TRANSACTION MANAGEMENT, CRASH RECOVERY AND CONCURRENCY CONTROL   (11 Periods)

Overview of Transaction Management: ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock Based Concurrency Control, Performance of Locking.

Concurrency Control: 2Phase Locking (2PL), Serializability and Recoverability, Lock Management, Lock Conversions.

Crash recovery: ARIES, the Log, Other Recovery related Structures, Write-Ahead Log Protocol, Check pointing, Recovering from a System Crash.

UNIT V - OVERVIEW OF STORAGE AND INDEXING   (11 Periods)

Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing: Clustered Indexes, Primary and Secondary Indexes; Index Data Structures: Hash based Indexing, Tree based Indexing.

Storing Data: Memory Hierarchy: Magnetic disks, Performance implications of disk structure; Redundant Arrays of Independent Disks.

Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. Raghu Ramakrishnan and Johannes Gehrke, "*Data base Management Systems*," Tata McGraw-Hill, 3rd Edition, 2013.

2. Ramez Elmasri, Shamkant B. Navathe, *"Database Systems*," Pearson Education, 6th Edition, 2013.

REFERENCE BOOKS:

1. A.Silberschatz, H.F. Korth, S.Sudarshan, "*Data base System Concepts*," McGraw hill, 6th Edition, 2011.

2. C.J.Date, "*Introduction to Database Systems,*" Pearson Education, 8th Edition, 2006.

ADDITIONAL LEARNING RESOURCES:

1. http://www.nptelvideos.in/2012/11/database-management-system.html

2. http://www.oracle.com/technetwork/tutorials/index.html

3. https://www.tutorialcup.com/dbms

4. https://www.javatpoint.com/oracle-tutorial

16MC20101: DATABASE MANAGEMENT SYSTEMS

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PREREQUISITES: --

COURSE DESCRIPTION:

Concepts of relational database and its design; Representation of ER diagram to Relational model; SQL queries; Normal forms; Recovery and concurrency control mechanism, Storage and indexing mechanism.

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

CO1. Gain in-depth knowledge in

- Database models and database architecture

- Transaction processing and recovery management

- Storage and Indexing mechanism

CO2. Analyze the complex problems of real world applications.

CO3. Design Relational Database Schema for a given Entity Relationship model.

CO4. Interpret the data by applying normalization techniques for the development of database application projects.

CO5. Use Structured Query Language DDL/DML/DCL commands to solve real time applications.

DETAILED SYLLABUS:

UNIT- I: INTRODUCTION TO CONCEPTUAL MODELING AND DATABASE DESIGN (11 Periods)

Introduction and Conceptual Modeling:

Database System Applications, database systems versus file systems, view of data: data abstraction, instances and schemas, data models: the entity-relationship model, relational models and other data models, database languages, database users and administrators, database system structure, history of database systems

Introduction to Database design: Database design and ER diagrams, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model.

UNIT – II RELATIONAL MODEL AND BASIC SQL   (10 Periods)

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design: ER to Relational, Introduction to Views, Destroying /altering Tables and Views.

Case Study: ER diagram for banking enterprise and university database

Basic SQL: SQL data definition and data types - CREATE table command in SQL, attribute data types in SQL, Specifying constraints in SQL- specifying attribute constraints and attribute defaults, specifying key and referential integrity constraints, specifying constraints on tuples using CKECK.

UNIT – III: SQL AND SCHEMA REFINEMENT AND NORMAL FORMS   (12 Periods)

SQL : Form of Basic SQL Query- Examples of Basic SQL Queries, Introduction to Nested Queries, correlated Nested Queries, Set- Comparison Operators, Aggregate Operators, NULL values-Comparison using Null values- Logical connectives- AND, OR and NOT- Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Triggers and Active Databases.

Schema Refinement and Normal Forms: Introduction to Schema Refinement- Problems Caused by redundancy, Decompositions, Problem related to decomposition, Functional Dependencies, Normal Forms – FIRST, SECOND, THIRD Normal forms, BCNF, Properties of Decompositions- Loss less- join Decomposition, Dependency preserving Decomposition.

UNIT - IV: OVERVIEW OF TRANSACTION MANAGEMENT, CRASH RECOVERY AND CONCURRENCY CONTROL   (11 Periods)

Overview of Transaction Management: The ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock Based Concurrency Control, Performance of Locking.

Concurrency Control: 2PL, Serializability and recoverability, Introduction to Lock Management, Lock Conversions.

Crash recovery: Introduction to ARIES, the Log, Other Recovery related Structures, the Write-Ahead Log Protocol, Check pointing, recovering from a System Crash.

UNIT - V: OVERVIEW OF STORAGE AND INDEXING   (11 Periods)

Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes, Index Data Structures- Hash based Indexing, Tree based Indexing.

Storing Data: The Memory Hierarchy- Magnetic disks, Performance implications of disk structure; Redundant Arrays of Independent Disks.

Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

Total Periods: 55

TEXT BOOKS

1. Raghu Ramakrishnan and Johannes Gehrke, "Data base Management Systems," Tata McGraw-Hill, 3rd Edition, 2007.

2. RamezElmasri, Shamkant B. Navathe, "Database Systems," Pearson Education, 6th Edition, 2013.

REFERENCE BOOKS

1. A.Silberschatz, H.F. Korth, S.Sudarshan, "Data base System Concepts," McGraw hill, 6th edition, 2006.

2. C. J. Date, "Introduction to Database Systems," Pearson Education, 7th Edition, 2004.

3. M. L. Gillenson, "Fundamentals of Database Management Systems," Wiley Student Edition, 2nd Edition, 2012.

4. Peter Rob and Carlos Coronel, "Database Systems Design, Implementation and Management," Cengage Learning, 8th Edition, 2009.

(19MC20104) **SOFTWARE ENGINEERING (Theory)**

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PRE-REQUISITES: --

COURSE DESCRIPTION:

Nature of Software, Unique Nature of WebApps, Software Engineering, Software Process, Software Engineering Practice; Process Models, Agile Development, Understanding Requirements, Requirements Modeling; Design Concepts, Architectural And Component-Level Design; User Interface Design and Testing Strategies; Product Metrics, Risk Management And Quality Management.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Identify and analyze software requirements, cost estimations, risk and quality management using the principles of software engineering, process models, design models, testing strategies, risk and management strategies to develop an application software.

CO2: Design software applications by applying design principles, user interface design rules, architectural design process, component-level design, agile development and metrics to develop a quality software product.

DETAILED SYLLABUS:

UNIT I - SOFTWARE AND SOFTWARE ENGINEERING, PROCESS MODELS AND AGILE DEVELOPMENT   (11 Periods)

Software and Software Engineering: Nature of Software, Unique Nature of WebApps, Software Engineering, Software Process, Software Engineering Practice, Software Myths. **Process Models:** Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology, Product and Process. **Agile Development:** Agility, Agility and Cost of Change, Agile Process, Extreme Programming, Other Agile Process Models.

UNIT II - UNDERSTANDING REQUIREMENTS AND REQUIREMENTS MODELING

(11 Periods)

Understanding Requirements: Requirements Engineering, Establishing the groundwork, Eliciting Requirements, Developing Use Cases, Building the requirements model, Negotiating Requirements, Validating Requirements.

Requirements Modeling (Scenarios, Information and Analysis Classes): Requirements Analysis, Scenario-Based Modeling, UML Models that Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling.

Requirements Modeling (Flow, Behavior, Patterns and WEBAPPS): Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model, Patterns for Requirements Modeling, Requirements Modeling for WebApps.

UNIT III- DESIGN CONCEPTS, ARCHITECTURAL AND COMPONENT-LEVEL DESIGN

(11 Periods)

Design Concepts: Design with Context of Software Engineering, Design Process, Design Concepts, Design Model.

Architectural Design: Software Architecture, Architecture Genres, Architecture Styles, Architectural Design, Assessing Alternative Architectural Designs, Architectural Mapping Using Data Flow.

Component-Level Design: Component, Designing Class-Based Components, Conducting Component-level Design, Component Level Design for WebApps, Designing Traditional Components, Component-Based Development.

UNIT IV- USER INTERFACE DESIGN AND TESTING STRATEGIES   (11 Periods)

User Interface Design: Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evaluation.

Testing Strategies: Strategic approach to software testing, test strategies for conventional software, Black-box and White-box testing, Validation testing, System testing, the art of debugging.

UNIT V - PRODUCT METRICS, RISK MANAGEMENT AND QUALITY MANAGEMENT

(11 Periods)

Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

Metrics for Process and Products: Software measurement, metrics for software quality.

Risk management: Reactive Vs proactive risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM plan.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. Roger S. Pressman, "*Software engineering A practitioner's Approach,*" Tata Mc-Graw Hill International Education, 7th Edition, 2016.
2. Ian Sommerville, "*Software Engineering,*" Pearson, 9th Edition, 2011.

REFERENCE BOOKS:

1. Rajib Mall, "*Fundamentals of Software Engineering,*" PHI, 5th Edition, 2018.
2. Hans Van Vliet, "*Software Engineering: Principles and Practices,*" Willey Publications, 3rd Edition, 2010.

ADDITIONAL LEARNING RESOURCES:

https://nptel.ac.in/courses/106101061

16MC20104: SOFTWARE ENGINEERING

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PREREQUISITES:- -

COURSE DESCRIPTION:

Software engineering core principles process models and agile process; design concepts and design issues; quality management principles; software configuration and product metrics; project estimation and risk management maintenance.

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

CO1. Understand concepts-process, models, methodologies and principles of software engineering.

CO2. Identify and classify user requirements and software requirement specification.

CO3. Analyze the architecture and Design application software using design engineering principles.

CO4. Estimate and maintain software configuration management by synthesis of development process to provide valid conclusions.

CO5. Apply risk and metrics management principles for quality assurance.

CO6. Test and communicate quality of an application and as per needs of the stakeholder.

DETAILED SYLLABUS:

UNIT - I: SOFTWARE PROCESS AND SOFTWARE PRINCIPLES   (12 periods)

The nature of Software , Software Myths, **Software Process Models:** A Generic Process Model, Process Models, **Core Principles** - Communication Principles, Planning Principles, Modeling Principles, Construction Principles, Deployment Principles.

Understanding Requirements: Requirements Engineering, Eliciting Requirements, Functional and non functional Requirements, Developing Use Cases, Validating Requirements, Introduction to Agile Processes - extreme Programming.

UNIT - II: SOFTWARE DESIGN AND SOFTWARE ARCHITECTURE   (12 periods)

Design concepts - Abstraction - Modularity - Refinement - **Architectural design** - Software Architecture, Architecture Design, Architecture Style, Architectural Mapping Using

Data Flow. **Component Level Design:** Component, Component Based Development. **User Interface Design:** The Golden Rules, Interface analysis and design, Interface design steps. Design Evaluation. **Web App Design:** Aesthetic, Content, Architecture, Navigation and Component level Design - Design issues.

UNIT - III: QUALITY MANAGEMENT PRINCIPLES   (11 periods)

Quality, Quality Control, Quality Assurance, **Review Techniques**-Informal Reviews, Formal Technical Reviews, Software Testing strategies, Unit Testing, Integration Testing, System Testing, Debugging Process, - Equivalence class Partitioning (ECP) , Boundary Value Analysis(BVA), White Box Testing, Black Box Testing. **Testing Web Applications:** Content, User Interface, Navigation, Configuration, Security and Performance Testing.

UNIT - IV: SOFTWARE CONFIGURATION MANAGEMENT AND PROJECT METRICS
                                                              (10 periods)

Software Configuration Management, **The SCM process:** Identification of objects, Version Control - Change control, Content Management, Change Management. **Software measurement** - - Size-oriented metrics and function point metrics .Object oriented Metrics, Use Case Oriented Metrics

UNIT V – PROJECT ESTIMATION, RISK MANAGEMENT AND MAINTENANCE

                                                              (10 periods)

Software Sizing, Problem Based Estimation Process Based Estimation, Estimation with Use Cases. Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring and Management, RMMM Plan. Software Maintenance , Software Supportability, Software Reengineering - Software reengineering process model - Reverse engineering to understand data, and processing

Total Periods: 55

TEXT BOOKS

1. Roger S. Pressman, "Software Engineering, A practitioner's Approach", McGraw-Hill International Edition, 7th edition, 2010.

REFERENCE BOOKS

1. K. K. Aggarwal & Yogesh Singh, "Software Engineering", New Age International Publishers, 3rd edition, 2007.

2. Shely Cashman Rosenblatt, "Systems Analysis and Design", Thomson Publications, 6th edition, 2006.

(19MC20131) **DATABASE MANAGEMENT SYSTEMS LAB**

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
50	50	100		-	-	3	1.5

PRE-REQUISITES:

A course on Database Management Systems

COURSE DESCRIPTION:

Creation of Data Definition commands; Implementation of Data Manipulation Commands; Table level and Column level Constraints; Construction of ER diagrams; Implementation of GROUP BY, HAVING, ORDER By clause; Creation and dropping of Views; Implementation of Nested Queries, Joins, Cursors, Functions, Procedures and Triggers.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Design and implement ER-diagrams, Relational schemas, table and column level constraints, Cursors, Triggers, Functions and procedures to develop Relational Database applications.

CO2: Select and apply Nested Queries, Joins, Views, Group functions in updating and managing the databases to solve relational database problems using SQL and PL/SQL languages.

CO3: Work independently and in team to solve problems with effective Communication.

LIST OF EXERCISES:

1. Creation of DDL Commands

Consider the following relation schemas:

Client_master(Client_no, Name, Address1, Address2, City, State, Pincode, Bal_due)

Product_master(Product_no, Description, Unit_measure, Qty_on_hand, Record_lvl, Sell_price, Cost_price)

Salesman_master(Salesman_id, Name, Address1, Address2, City, State, Pincode, Sal_amt, Target_amt, Remarks)

Create the above tables, insert suitable tuples and perform the following DDL operations (CREATE, ALTER, DROP, RENAME, TRUNCATE) in Oracle SQL.

2. Creation of DML Commands

Using the table client master and product master answer the following Questionnaires.

i. Change the selling price of '1.44 floppy drive to Rs.1150.00

ii. Delete the record with client 0001 from the client master table.

iii. Change the city of client_no'0005' to Bombay.

iv. Change the bal_due of client_no '0001, to 1000.

v. Find the products whose selling price is more than 1500 and also find the new selling price as original selling price *15.

vi. Find out the clients who stay in a city whose second letter is a.

vii. Find out the name of all clients having 'a' as the second letter in their names.

viii. List the products in sorted order of their description.

ix. Count the total number of orders

x. Calculate the average price of all the products.

xi. Calculate the minimum price of products.

xii. Determine the maximum and minimum prices . Rename the tittle as 'max_price' and min_price respectively.

xiii. Count the number of products having price greater than or equal to 1500.

3. Creation of Table level and Column level Constraints

Implement table level and column level constraints like Domain Integrity constraints (NOT NULL, CHECK), Entity integrity constraints (UNIQUE, PRIMARRY KEY) and Referential integrity constraints (FOREIGN KEY) for the below tables.

Create table Salesman_master with the following columns and constraints-

Column name	Data type	Size	Attributes
Salesman_no	Varchar2	6	Primary key/first letter must start with 's'
Sal_name	Varchar2	20	Not null
Address	Varchar2	20	Not null
City	Varchar2	20	
State	Varchar2	20	
Pincode	Number	6	
Sal_amt	Number	8,2	Not null, cannot be 0
Tgt_to_get	Number	6,2	Not null, cannot be 0
Ytd_sales	Number	6,2	Not null, cannot be 0
Remarks	Varchar2	30	

Create table sales_order with following columns and constraints

Column name	Data type	Size	Attributes
S_order_no	Varchar2	6	Primary/first letter must be 0
S_order_date	Date	6	Primary key reference client_no of client_master table
Client_no	Varchar2	25	
Dely_add	Varchar2	6	

Salesman_no	Varchar2	6	Foreign key references salesman_no of salesman_master table
Dely_type	Char	1	Delivery part(P)/full(F), default F
Billed_yn	Char	1	
Dely_date	Date		Can not be lessthan s_order_date
Order_status	Varchar2	10	Values ('In Process'; 'Fulfilled'; Back Order'; Canceled

Create table sales_order_details with following columns and constraints

Column name	Data type	Size	Attributes
S_order_no	Varchar2	6	Primary key/foreign key references s_order_no of sales_order
Product_no	Varchar2	6	Primary key/foreign key references product_no of product_master
Qty_order	Number	8	
Qty_disp	Number	8	
Product_rate	Number	10,2	

4. Construction of ER diagram

a) Construct an ER diagram for a **University** database application. Identify the Relations and include necessary integrity constraints.

An University has many departments, where each department has multiple Instructors. An Instructor belongs to only one department. Each department offers multiple Courses, each of which is taught by a single Instructor. A student may enroll for many courses offered by the department.

For the above problem create the following:

   i. Analyze the date required

   ii. Normalize the attributes

   iii. Create the logical data model (ER diagram)

b) Construct an ER diagram for a **Bank Database** by considering the following set of requirements that is used to keep track of Customer.

   i.   Each bank has a unique name.

   ii. Each branch has a number, name, address (number, street, city), and set of phones.

   iii. Customer includes their name, set of address (P.O. Box, city, zip code, country), set of phones, and social security number.

   iv. Accounts have numbers, types (e.g. saving, checking) and balance. Other branches might use the same designation for accounts. So to name an

24

account uniquely, we need to give both the branch number to which this account belongs to and the account number.

     v.   Not all bank customers must own accounts and a customer may have at most 5 accounts in the bank.

    vi.   An account must have only one customer.

   vii.   A customer may have many accounts in different branches.

5. Group functions: Implement queries using Aggregate functions, GROUP BY, HAVING, ORDER By clause and Creation, dropping of Views for the above tables using SQL.

6. Subqueries: Implement queries using SUBQUERIES for the above tables using SQL.

7. Joins: Implement queries using JOINS and OUTER JOINS for the above tables using SQL.

PL/SQL PROGRAMS:

8. PL/SQL **CURSOR** program

Write a PL/SQL program for generation of Electricity Bill using CURSORS.

    Note: Create a table for Electricity bill consists of Customer_no, Customer_name, Customer_type, Prev_met_read, Curr_met_read, Month_name. Assume there are three Customer types namely Industrial, Agriculture and Domestic. Calculate the total charges based on the type of customer.

9. Triggers

Generate a database trigger to update the salary of an employee before/after performing any DML operations.

10. Procedures and Functions

    a) Write a procedure which takes the department_id as an input parameter and lists the names of all employees belonging to that department.

    b) Write a PL/SQL block of code that lists the highest salary drawn by an employee in each of the departments. It should make use of a function dept_highest which return the highest salary drawn by an employee for the given department.

Mini Project 1: Consider the following schema for a **Library Database:**

    BOOK(Book_id, Title, Publisher_Name, Pub_Year)

    BOOK_AUTHORS(Book_id, Author_Name)

    PUBLISHER(Name, Address, Phone)

    BOOK_COPIES(Book_id, Branch_id, No-of_Copies)

    BOOK_LENDING(Book_id,   Branch_id,   Card_No,   Date_Out,   Due_Date)

    LIBRARY_BRANCH(Branch_id, Branch_Name, Address)

Implement the following SQL queries to:

    a) Retrieve details of all books in the library–id, title, name of publisher, authors, number of copies in each branch, etc.

b) Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.
c) Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
d) Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.
e) Create a view of all books and its number of copies that are currently available in the Library.

Mini Project 2: Consider the schema for **College Database**:

STUDENT(USN, SName, Address, Phone, Gender)

SEMSEC(SSID, Sem, Sec)

CLASS(USN, SSID)

SUBJECT(Subcode, Title, Sem, Credits)

IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)]

Implement the following SQL queries to:

a) List all the student details studying in fourth semester 'C' section.
b) Compute the total number of male and female students in each semester and in each section.
c) Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.
d) Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.
e) Categorize students based on the following criterion: If FinalIA = 17 to 20 then CAT = 'Outstanding' If FinalIA = 12 to 16 then CAT = 'Average' If FinalIA< 12 then CAT = 'Weak' Give these details only for 8th semester A, B, and C section students.

REFERENCE BOOKS:

1. Ivan Bayross, *"SQL, PL/SQL The Programming Language of ORACLE,"* BPB Publications, 2010.

2. Dr. P. S. Deshpande, *"SQL & PL/SQL for Oracle 10g Black Book,"* Dreamtech Press,

2011.

3. J. J. Patrick, *"SQL Fundamentals,"* Pearson Education, 3rd Edition, 2008.

4. Rick F. Vander Lans*, "Introduction to SQL,"* Addison-Wesley Professional, 4th Edition, 2006.

16MC20131: DATABASE MANAGEMENT SYSTEMS LAB

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	3	2

PREREQUISITES:A Course on "Database Management Systems".

COURSE DESCRIPTION:

Analyze problems and design of ER diagrams; Creation of Data Definition commands; Normalization techniques; Implementation of functions; Creation of Views, Indexes and Sequences; Implementation of simple and complex queries using Oracle SQL; Creation of packages and triggers.

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

CO1. Impart knowledge in applying normalization techniques for development of application software to realistic problems.

CO2. Formulate queries using SQL DDL/DML/DCL commands.

CO3. Design a database using ER diagrams, convert ER diagrams into relation schemas.

CO4. Improve the database performance by optimizing the queries using Indexing and Hashing techniques.

CO5. Exploit their knowledge in developing database applications using SQL language.

LIST OF EXERCISES

1. **Creation of DDL Commands**

   Using SQL, create simple DDL Commands (CREATE, ALTER, DROP, RENAME, TRUNCATE) on the following schema of tables.

   EMP (Empno number (5), Ename varchar2 (20), Sal number (8, 2), Designation varchar2 (20), Address varchar2 (20))

   DEPT (Deptno number (3), Dname varchar2 (20), Location varchar2 (25))

2. **Creation of DML Commands**

   Implement various DML commands and execute simple SQL queries.

3. **Creation of Table level and Column level Constraints**

Implement table level and column level constraints like Domain Integrity constraints (NOT NULL, CHECK), Entity integrity constraints (UNIQUE, PRIMARRY KEY) and Referential integrity constraints (FOREIGN KEY).

4. Retrieving of data using comparison operators and logical operators

Practice of simple SQL queries using comparison operators (=, !=, >, <, >=, <=, <>, between, in, not in, null) and logical operators(and, or, not).

5. ER diagram for an University database

Construct an ER diagram for a University database application. Identify the Relations and include necessary integrity constraints.

An University has many departments, where each department has multiple Instructors. An Instructor belongs to only one department. Each department offers multiple Courses, each of which is taught by a single Instructor. A student may enroll for many courses offered by the department.

Implement the following queries:

i. Find the names of all the students whose total credits are greater than 100

ii. Find the course-id and grades of all courses taken by any student named 'Tanaka'

iii. Find the ID and name of instructors who have taught a course in the Computer Science department.

iv. Find the courses which are offered in both 'Fall' and 'Spring' semester (not necessarily in the same year).

v. Find the names of all the instructors from Computer Science department.

vi. Find the course-id and titles of all courses taught by an instructor named 'Srinivasan'

vii. Find names of instructors who have taught at least one course in Spring 2009

6. Single Row Functions

Implement queries using Single row functions such as Numeric functions, Date functions, Conversion functions and String functions.

7. Group functions

Practice Group functions such as Sum, Avg, Max, Min, Count.

8. Group by Having Clause

Implement SQL queries using Group By and Having Clause.

9. Creation of VIEWS

Creation of tables using Simple View and Complex View

10.Synonym and Sequences

Implementation of Synonym and Sequences.

11.JOINS

Practice queries using JOINS and OUTER JOINS.

12.SUBQUERIES

Implementation queries using SUBQUERIES.

13. PL/SQL basic programs

a) Write a simple PL/SQL program to accept a number from user and test whether it is divisible by a number

b) Write a PL/SQL program to check whether the input is a character, number or a special character.

c) Write a PL/SQL code to update the salary of an employee based on given bonus and department number.

14. PL/SQL CURSOR programs

Write a PL/SQL program for generation of Electricity Bill using CURSORS.

Create a table for Electricity bill consists of Customer_no, Customer_name, Customer_type, Prev_met_read, Curr_met_read, Month_name. Assume there are three Customer types namely Industrial, Agriculture and Domestic. Calculate the total charges based on the type of customer.

15.   Triggers

Generate a database trigger to update the salary of an employee before/after performing any DML operations.

16.   Procedures

Write a procedure which takes the department_id as an input parameter and lists the names of all employees belonging to that department.

17.   Functions

Write a Pl/Sql block of code that lists the highest salary drawn by an employee in each of the departments. It should make use of a function dept_highest which return the highest salary drawn by an employee for the given department.

18.   Packages

Create a package to find the salary of an Employee by providing   employee_id as an argument.

REFERENCE BOOKS

1. Ivan Bayross, *"SQL, PL/SQL The Programming Language of ORACLE,"* BPB Publications, 2002.

2. Dr. P. S. Deshpande, *"SQL & PL/SQL for Oracle 10g Black Book,"* Dreamtech Press, 2007.

3. J. J. Patrick, *"SQL Fundamentals,"* Pearson Education, 2nd Edition, 2002.

4. Rick F. Vander Lans,*"Introduction to SQL*," Addison-Wesley Professional, 4th Edition, 2007.

(19MC20132) **DATA STRUCTURES LAB**

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
50	50	100		-	-	3	1.5

PRE-REQUISITE:

Courses on Programming in C Lab and Data Structures.

COURSE DESCRIPTION:

Logical and physical representation of data, Abstract data types of Linear and Non-linear Data structures; Singly and Doubly Linked lists; Queues, Stacks and their applications; Binary tree, Binary Search Tree and AVL trees; Usage of graphs, Sorting and Searching techniques.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Analyze and implement algorithms to perform the operations of stacks, queues, linked lists, trees and graphs using problem solving approaches to solve computational problems.

CO2: Develop programs using sorting and searching techniques, trees and graph traversal techniques to solve memory utilization problems.

CO3: Work independently and in a team to solve problems with effective Communication.

LIST OF EXERCISES:

1. Write a C program to implement Stack and Queues using pointers.

2. Write a C program to perform the following expression using Stack operations

    a) Convert infix expression into postfix expression

    b) Evaluate postfix expression

3. Write a C program to implement the following:

    a) Singly linked list

    b) Doubly Linked List

4. Write a C program to implement Towers of Hanoi using Stack.

5. Write a C program to implement the following sorting algorithms:

    i) Bubble sort   ii) Selection sort   iii) Quick sort   iv) Merge sort

6. Write a C program that use both Recursive and non-recursive functions to perform the following searching operations:

    i) Linear search   ii) Binary search

7. Write a C program to create Binary Search Tree and perform operations on it.

8. a) Write a C program to implement Recursive Tree traversal techniques.

b) Write a C program to implement non-recursive Tree traversal techniques.

9. Write a C program to create AVL-tree and perform operations on it.

10. Write a C Program to implement Prim's Algorithm.

Mini Project-1:

Design, Develop and Implement a Program in C for the following operations on Graph of Cities
a. Create a Graph of N cities using Adjacency Matrix.
b. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method.

Mini Project-2:

Design, Develop and Implement a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo.

Perform the following operations on Employee data:

a. Create a DLL of N Employees Data by using end insertion.

b. Display the status of DLL and count the number of nodes in it

c. Perform Insertion and Deletion at End of DLL

d. Perform Insertion and Deletion at Front of DLL

e. Demonstrate how this DLL can be used as Double Ended Queue

f. Exit

REFERENCE BOOKS:

1. P. Padmanabham, "*C programming and Data Structures,*" B.S. Publications, 3rd Edition, 2008.

2. M.T. Somashekara, D. S. Guru, K. S. Manjunatha, "*Problem Solving with C,*" PHI Learning, 2nd Edition, 2018.

3. E. Karthikeyan, "*A Textbook on C: Fundamentals, Data Structures and Problem Solving,*" Prentice Hall of India Private Limited, 2008.

ADDITIONAL LEARNING RESOURCES:

1. https://nptel.ac.in/courses/106102064
2. https://nptel.ac.in/courses/106103069
3. https://nptel.ac.in/courses/106102064/24
4. https://swayam.gov.in/course/235-data-structure

16MC20132: DATA STRUCTURES THROUGH C LAB

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
50	50	100		-	-	3	2

PREREQUISITES: Courses on "Programming in C Lab" and "Data Structures".

COURSE DESCRIPTION

Implementing logical and physical representation of data, complexity and their efficiency. Implementing linked lists and their different variations, queues, stacks and their applications; tree structures and their different variations; Solving problems using graphs, sorting and searching techniques.

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

CO1. Apply abstract data type and their basic usability in different applications through C programming language.

CO2. Identify and analyze suitable data structures to solve computing problems.

CO3. Design and develop variety of c programs using data structures in order to solve computing problems.

CO4. Choose the appropriate data structure and algorithm design method to get an optimal solution for complex real world problem.

CO5. Apply searching, sorting, tree traversal and graph traversal techniques to optimize the complexities of an application.

CO6. Work together or as an individual to customize the applications.

LIST OF EXERCISES

6. Write C programs that implement stack and its operations using

   a) Arrays   b) Pointers

7. Write C programs that uses Stack operations to perform the following:

   a) Converting infix expression into postfix expression

   b) Evaluating the postfix expression

8. Write C programs that implement Queue and its operations using

   a) Arrays   b) Pointers

9. Write a C program that uses functions to perform the following operations on single linked   list:

   i) Creation   ii) Insertion

   iii) Deletion   iv) Traversal

10. Write a C program that uses functions to perform the following operations on double linked list

   i) Creation   ii) Insertion

   iii) Deletion   iv) Traversal

6. Write a C program that uses functions to perform the following operations on Circular linked list

   i) Creation   ii) Insertion

   iii) Deletion   iv) Traversal

7. Write a C program that implements the following sorting methods to sort a given list of integers in ascending order

   i) Bubble sort   ii) Selection sort

8. Write a C program that implements the following sorting methods to sort a given list of integers in ascending order

   i) Quick sort   ii) Merge sort

9. Write C programs that use both recursive and non recursive functions to perform the following searching operations for a Key value in a given list of integers:

   i) Linear search   ii) Binary search

10.   Write a C program to create Binary Search Tree and perform operations on it.

11. a) Write a C program to implement recursive Tree traversal techniques.

   b) Write a C program to implement non-recursive Tree traversal techniques.
12. Write a C program to create AVL-tree and perform operations on it.
13. Write a C program to implement Heap Sort.
14. Write a C program to implement Graph traversal Techniques (BFS, DFS)
15. Write a C Program to implement Prim's Algorithm

REFERENCE BOOKS

1. P. Padmanabham, "*C programming and Data Structures,*" BS Publications, 3rd Edition, 2008.
2. M.T. Somashekara, "Problem Solving with C", PHI Learning Private Limited: New Delhi, 2012.
3. E. Karthikeyan, "A Textbook on C Fundamentals, Data Structures and Problem Solving," Prentice Hall of India Private Limited: New Delhi, 2008.

(19MC30102) **DATA WAREHOUSING AND DATA MINING (Theory)**

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PRE-REQUISITE: --
A Course on Database Management Systems.

COURSE DESCRIPTION:

Data Warehouse Components; Building Data Warehouse; Data mining; Data Preprocessing; Association Rule Mining; Classification and Clustering techniques; Mining different types of data and its Applications.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Design and develop data warehouse architecture, multidimensional schemas to perform business analysis using OLAP tools.

CO2: Select and apply association rule mining and classification algorithms to identify the frequent patterns and predictions effectively.

CO3: Apply Clustering algorithms on preprocessed datasets to find solutions of real time applications.

CO4: Analyze data preprocessing methods and data mining functionalities to mine text, multimedia, web and spatial data to discover knowledge.

DETAILED SYLLABUS:

UNIT I- DATA WAREHOUSE COMPONENTS   (11 periods)

DATA WAREHOUSE COMPONENTS: Data Warehousing, Paradigm Shift, Business Problem Definition, operational and informational Data Stores, Data Warehouse Definition and Characteristics, A Multi-tiered Data Warehouse Architecture. Meta data, data marts, Data Warehouse Administration and Management, Benefits of Data Warehousing, Multidimensional Data Model-From tables and spread sheets to Data Cubes and Star, Snowflake and fact constellation Schemas, Role of Concept hierarchies, Measures, OLAP Operations, From online Analytical processing to Multidimensional Data Mining.

UNIT II- DATA MINING AND DATA PREPROCESSING   (11 periods)

DATA MINING: Motivated Data Mining, Kinds of Data, Data mining Functionalities, classification of Data mining systems, Data mining primitives, Integration of Data mining Systems with a Database or Data Warehouse System, Major issues in Data Mining.

DATA PREPROCESSING: Need for Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT III- ASSOCIATION RULE MINING AND CLASSIFICATION   (12 periods)

ASSOCIATION RULE MINING: Basic Concepts, The Apriori algorithm for finding frequent itemsets using candidate generation, Generating association rules from frequent itemsets, Mining frequent itemsets without candidate generation.

CLASSIFICATION: Classification, issues in classification and prediction, Classification by Decision Tree Induction, Bayesian Classification, Rule-based classification, Prediction: Linear Regression, Accuracy and Error measures, evaluating the accuracy of a classifier or predictor, bagging and boosting.

UNIT IV- CLUSTER ANALYSIS   (11 periods)

CLUSTERING: Cluster Analysis**,** Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning methods - k-means and k-mediods methods, CLARANS, Hierarchical Methods**-**Agglomerative and divisive hierarchical clustering, Density-Based Method-DBSCAN, Grid-Based Method-STING, Outlier Analysis.

UNIT V- MINING DIFFERENT TYPES OF DATA AND ITS APPLICATIONS

                                                              (10 periods)

MINING DIFFERENT TYPES OF DATA: Multimedia Data Mining, Text Mining - Text data analysis and informational retrieval, text mining approaches, Spatial Mining, Mining the World Wide Web- Mining web page layout structure, Mining web's link structures, Web usage mining.

DATA MINING APPLICATIONS: Financial data Analysis, Retail Industry, Telecommunication Industry.

                                                              Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. Jiawei Han, Micheline Kamber and Jian Pei, Data Mining: Concepts and Techniques, Elsevier, 3rd Edition, 2013.

2. Berson Alex and Stephen J Smith, *"Data Warehousing, Data Mining and OLAP,"* Tata McGraw-Hill, 2004.

REFERENCE BOOKS:

1. Ralph Kimball, Margy Ross, Warren Thornthwaite and Joy Mundy, Bob Becker, *"The Data Warehouse Life cycle Tool kit,"* John Wiley and Sons Inc, 2nd Edition, 2007.

2. William H Inmon, *"Building the Data Warehouse,"* John Wiley and Sons Inc, 4th Edition, 2005.

3. Arun K Pujari, *"Data Mining Techniques,"* Universities Press (India) Pvt. Ltd, 2nd Edition, 2001.

4. G. K. Gupta, *"Introduction to Data Mining with Case Studies,"* Easter Economy Edition, Prentice Hall of India, 2006.

ADDITIONAL LEARNING RESOURCES:

1. https://nptel.ac.in/courses/106/105/106105174/
2. https://www.youtube.com/watch?v=vuc93jbO2Dw.
3. Data Mining - Concepts and Techniques (3rd edition) by Jiawei Han, Micheline Kamber and Jian Pei.
4. Tutorial on Data Mining Algorithms by Ian.
5. *Mining of Massive Datasets* by Anand Rajaraman and Jeff Ullman.
6. Open source tools for data mining:http://eprints.fri.uni-lj.si/893/1/2008-OpenSource DataMining.pdf

16MC30102: DATA WAREHOUSING AND DATA MINING

Int. Marks	Ext.Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A Course on "Data Base Management Systems".

COURSE DESCRIPTION:

Data Warehouse Components and Architecture; Data mining Functionalities; Data Preprocessing; Association Rule Mining; Classification and Clustering; Multimedia, Text, Web Data Mining and Applications.

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

CO1. Demonstrate knowledge on:

- Schemas of Data warehouse
- Data preprocessing methods
- Classification and Clustering techniques

CO2. Analyze frequent itemsets using Apriori and FP growth algorithms.

CO3. Design and develop solutions for different classification and prediction models.

CO4. Solve complex problems by adapting appropriate analysis and interpretation of different types of text, multimedia and web data.

CO5. Use WEKA tool for creation of weather, hospital, banking dataset and perform preprocessing on these datasets.

DETAILED SYLLABUS:

UNIT-I: INTRODUCTION AND DATA WAREHOUSE COMPONENTS

(10 periods)

Introduction: The need for Data Warehousing, Paradigm Shift, Business Problem Definition, operational and informational Data Stores, Data Warehouse Definition and Characteristics, Data Warehouse Architecture.

Data Warehouse Components: Overall Architecture, Data Warehouse Database, Sourcing, Acquisition , Cleanup and Transformation tools, meta data, data marts, Data Warehouse Administration and Management .

UNIT-II: BUILDING A DATA WAREHOUSE AND INTRODUCTION TO DATAMINING

(12 periods)

Building A Data Warehouse: Business Consideration, Design considerations, Technical considerations, Implementation considerations, integrated solutions, Benefits of Data Warehousing, Multidimensional Data Model-From tables and spread sheets to Data Cubes and Star, Snowflake and fact constellation Schemas

Introduction to Data Mining: Motivated Data Mining, Definition of Data Mining, Kinds of Data, Data mining Functionalities, classification of Data mining systems, Data mining primitives, Integration of Data mining Systems with a Database or Data Warehouse System, Major issues in Data Mining.

UNIT-III: DATA PREPROCESSING AND ASSOCIATION RULE MINING

(12 periods)

Data Preprocessing: Need for Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

Association Rule Mining: Basic Concepts, The Apriori algorithm for finding frequent itemsets using candidate generation, Generating association rules from frequent itemsets, Mining frequent itemsets without candidate generation.

UNIT-IV: CLASSIFICATION AND CLUSTERING   (11 periods*)*

Classification: Definition of classification, Definition of prediction, issues in classification and prediction, Classification by Decision Tree Induction, Bayesian Classification, Accuracy and Error measures, evaluating the accuracy of a classifier or predictor, bagging.

Clustering: Introduction to cluster Analysis**,** Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning methods - k-means and k-mediods methods, CLARANS, Hierarchical Methods**-**Agglomerative and divisive hierarchical clustering.

UNIT-V: MULTIMEDIA, TEXT AND WEB DATA MINING APPLICATION   (10 periods*)*

Mining different types of data: Multimedia Data Mining, Text Mining - Text data analysis and informational retrieval, text mining approaches, Mining the World Wide Web- Mining web page layout structure, Mining web's link structures, Web usage mining.

Data Mining Applications: Financial data Analysis, Retail Industry, Telecommunication Industry.

Total Periods: 55

TEXT BOOKS:

1. Jiawei Han, Micheline Kamber and Jian Pei, *"Data Mining–Concepts and Techniques,"* Morgan Kaufmann Publishers, 2nd Edition,2006

2. Berson Alex and Stephen J Smith, *"Data Warehousing, Data Mining and OLAP,"* Tata McGraw-Hill, 2004.

REFERENCE BOOKS:

1. Ralph Kimball, Margy Ross, Warren Thornthwaite and Joy Mundy, Bob Becker, *"The Data Warehouse Life cycle Tool kit,"* John Wiley & Sons Inc, 2nd Edition, 2007.

2. William H Inmon, *"Building the Data Warehouse,"* John Wiley & Sons Inc, 4th Edition, 2005.

3. Arun K Pujari, *"Data Mining Techniques,"* Universities Press (India) Pvt. Ltd, 2nd Edition, 2001.

(19MC30103) PYTHON PROGRAMMING (Theory)

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		3	1	-	4

PRE-REQUISITES:

Courses on Programming in C and Object Oriented Programming through JAVA.

COURSE DESCRIPTION:

Data types and Expressions; Control Statements; Strings; Text Files; Lists; Dictionaries; Functions; Objects and their use; Exception Handling; Design with Classes; Graphical user Interface.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Understand the concepts of computer algorithms, problem solving, data types, control structures, expressions, lists, dictionaries, and tuples.

CO2: Design functions, modules, algorithms to solve recursive and non recursive problems.

CO3: Apply python programming constructs, Objects, strings and text files to implement Object Oriented Programming applications using Jupyter Notebook.

CO4: Implement towers of hanoi, Cigarette Use/ Lung cancer Correlation programs using Dictionaries, list, sets, tuples and functions.

CO5: Develop GUI based applications using tkinter, Python programming and object-oriented programming constructs to solve problems.

DETAILED SYLLABUS:

UNIT I- PYTHON PROGRAMMING CONCEPTS   (10 Periods)

Computer science and problem solving, Computer and algorithms a perfect match, Computer software-Syntax, semantics and problem translation, the process of computational problem solving, The Python programming language, python development environment, basics of python, First program in Python-calculating the drake equation.

Data Types and Expressions: Literals, Variables and Identifiers, Operators, Expressions and Data types, program to calculate individual's age in seconds.

UNIT II- CONTROL STRUCTURES, LISTS, DICTIONARIES, TUPLES AND SETS

(9 Periods)

Control Structures: Control structures, Boolean expressions, Selection control and Iterative control.

Lists: List structures, Lists in Python, Iterations over lists, Assigning and copying lists, List comprehensions.

Dictionaries, Tuples and Sets: Dictionary types, Implementation of Dictionary, Tuples, Sets, program to add and access elements from the dictionary.

UNIT III- DESIGN FUNCTIONS, MODULES, STRINGS AND TEXT FILES   (8 Periods)

Functions: Program routines, Functions.

Recursion: Recursion-Recursive functions, Recursive problem solving, comparing Iteration with Recursion.

Case study 1: Towers of Hanoi using recursion (The Towers of Hanoi problem is based on a legend of unknown origin. According to the legend, there is a Vietnamese temple with a large room containing three pegs and 64 golden disks. Each disk has a hole in it so that it can be slipped onto any of the pegs. In addition, each disk is of different size. The 64 disks are moved by priests from one peg to another with the following conditions:

- Only one disk can be moved at a time.
- At no time can a larger disk be placed on top of a smaller one.)

Text Files: Using text files, String processing, Exception handling.

Module Design: modules, top-down design, Python modules.

Case study 2: Cigarette Use/ Lung cancer Correlation program. (Computing the correlation between Cigarette use/ Lung cancer)

UNIT IV- OBJECTS AND OBJECT ORIENTED PROGRAMMING   (9 Periods)

Objects: Software objects, Turtle graphics- Creating a turtle graphics window, default turtle, Fundamental turtle attributes and behavior, Additional turtle attributes, creating multiple turtles.

Case Study: Horse Race Simulation problem (design, implement and test a program that simulates a horse race)

Object Oriented Programming: concepts of object oriented programming, Encapsulation, Inheritance and Polymorphism.

UNIT V- GRAPHICAL USER INTERFACE PROGRAMMING   (9 Periods)

Tkinter Overview - tkinter pragmatics, Documentation, Extensions, structure; making widgets; tkinter coding alternatives, configuring widgets, adding buttons and callbacks-lambda, bound method, callable class object, Binding events; adding multiple widgets, Reusable GUI Components with classes, Dialogs, Entry, check buttons, Radio buttons, Scales and Menus.

Case Study: Create a Canvas with options, where user presses the button, it should draw a required shape on the canvas*.*

Total Periods: 45

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. Charles Dierbach, "*Introduction to Computer Science using Python: A Computational Problem-Solving Focus,"* Wiley India Edition, 2016.

2. Mark Lutz, "*Programming Python*," O'Reilly Publications, 4th Edition, 2011.

REFERENCE BOOK:

1. Kenneth Lambert and B.L. Juneja, "*Fundamentals of Python*, Cengage Learning," 3rd Edition, 2012.

ADDITIONAL LEARNING RESOURCES:

1. https://docs.python.org/3/tutorial/
2. https://pythonprogramming.net/introduction-to-python-programming/

MCA III- SEMESTER
(19MC30131) **COMPUTER NETWORKS LAB**

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	3	1.5

PRE-REQUISITES:

A course on Computer Networks.

COURSE DESCRIPTION:

Implementing error detection and correction techniques; sliding window protocol; simulation of routing algorithms; congestion controlling mechanism; implementation of various Transport layer protocols.

COURSE OUTCOMES:

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

CO1: Understand guided and unguided media, Network Configurations, Network topologies, and networking devices.

CO2: Apply network simulation tools to simulate routing algorithms, flow control techniques to provide solutions in compliance with networking standards.

CO3: Implement client-server Communication, RMI, TCP/IP sockets, transport layer protocols, Congestion control protocols, error detection and correction mechanisms following Networking Principles and Standards.

CO4: Function effectively as an individual to solve problems with effective Communication.

CO5: Write and present a technical report / document effectively.

LIST OF EXERCISES:

1.   a)   Study of different types of network cables and implement cross wired cable and straight through cable using clamping tool

   b)   Study of network devices and network IP in detail.

   c)   Construct a peer to peer topology.

2.   a)   Study of basic network commands and network configuration commands.

          i) ping ii) nslookup iii) netstat iv) ipconfig

   b)   Create and configure a network topology with four PCs, two switches, and two routers.

3.   Implement the Data Link layer error detecting method using CRC-CCITT

(16-bits).

4.   Implement the Data Link layer error detection and correction techniques using Hamming Code.

5.   a)   Introduction to Simulation tool (NS2/NS3).

       b)   Create a network with three nodes namely 0, 1 and 2. Establish a TCP connection between node 0 and node 2 such that node 0 will send TCP packets to node 2 via node 1.

6.   a)   Sliding window protocol supports reliable and efficient transmission between nodes and it also obtains higher throughput than that of stop-n-wait protocol. Simulate sliding window protocol normal operation and timeout operations.

       b)   Implement the congestion control using Leaky bucket algorithm.

7.   Configure the following network to find shortest path between R2 LAN to R3 LAN using Distance Vector/Link State Routing Protocol. Each path is labeled with an arbitrary value for cost.

Shortest Path for host on R2 LAN to reach host on R3 LAN:
R2 to R1 (20) + R1 to R3 (5) + R3 to LAN (2) = 27
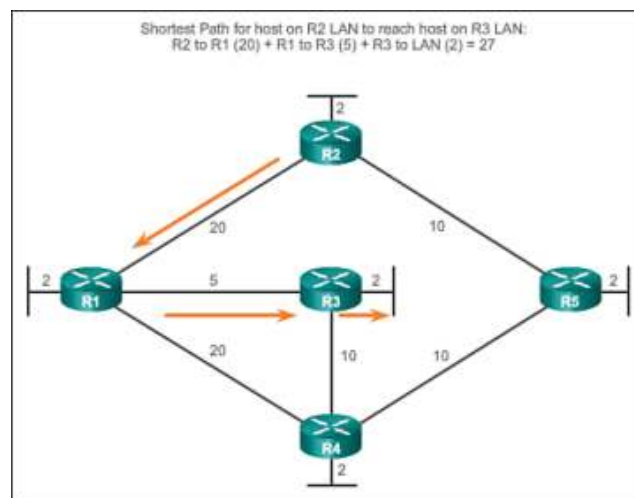
8.   a)   Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

       b)   In a typical FTP session, the user is sitting in front of one host (the local host) and wants to transfer files to or from a remote host. Implement File

9. Implementation of Socket Programming using UDP.

10. a) Implement Remote Procedure Call (RPC) using Remote Command Execution.

    b) Develop a code for Remote Method Invocation (RMI).

11. Write a program to simulate Address Resolution Protocol (ARP) and Reverse Address Resolution Protocols (RARP) used in Transport Layer.

12. a) Design a simple textual chat application that resembles Talk command in UNIX.

    b) Implementation of ping server and client application using sockets.

REFERENCE BOOKS:

1. Andrew S. Tanenbaum and David J. Wetherall, *"Computer Networks,"* Pearson Education, 5th Edition, 2015.
2. Behrouz Forouzan, "*Introduction to Data Communications and Networking",* Tata McGrawHill, 5th Edition, 2015.

SOFTWARE/TOOLS TO BE USED:

1. Network simulator tools – NS2/NS3, Packet Tracer
2. Virtual Labs
3. (Computer Networks Lab – http://vlabs.iitb.ac.in/vlabs-dev/labs_local/computer-networks/labs/explist.php)
4. Virtual Labs
5. (Advanced Network Technologies Virtual Lab - http://vlabs.iitkgp.ernet.in/ant/)

ADDITIONAL LEARNING RESOURCES:

1. https://www.itprc.com/packet-tracers/

2. https://www.nsnam.org/docs/tutorial/html/

3. http://www.tcpipguide.com/free/t_OSIReferenceModelLayers.htm

MCA – III Semester

16MC30131: COMPUTER NETWORKS LAB

Int. Marks	Ext.Marks	Total Marks	L	T	P	C
50	50	100	-	-	3	2

PREREQUISITES: A Course on "Computer Networks".

COURSE DESCRIPTION:

Implementing error detection and correction techniques; sliding window protocol; simulation of dynamic routing algorithms; congestion controlling mechanism; simulation of various Transport layer protocols.

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

CO1. Demonstrate knowledge on the concepts of networks, topologies, network devices and network simulators.

CO2. Analyze Error detection and correction mechanisms to verify and correct the data.

CO3. Develop networking protocols like TCP/IP, UDP, RPC, ARP and RARP.

CO4. Investigate congestion control mechanisms such as Leaky Bucket algorithm to achieve flow control.

CO5. Simulate dynamic routing protocols such as Distance Vector and Link state routing algorithms using NS2 simulator.

CO6. Adapt policies and mechanisms to avoid unauthorized access over the network through access control mechanisms and authentication.

CO7. Effectively communicate the routing paths through network simulators through visualization.

CO8. Advanced communication techniques can be applied by an individual to interact with remote machine through client server programming.

LIST OF EXERCISES:

1. Implement the Data Link layer error detecting method using CRC-CCITT (16-bits).
2. Implement the Data Link layer error detection and correction techniques using Hamming Code.
3. Simulate the Sliding Window Protocols used in Data Link layer to achieve flow control.
4. Simulate the congestion control using Leaky bucket algorithm.
5. Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.
6. Implementation of Socket Programming using UDP.
7. Write a program to implement subnetting and find the subnet masks.

8. a) Write a program to implement Remote Procedure Call using Remote Command Execution.

   b) Implementation of RMI.

9. Write a program to simulate Address Resolution Protocol (ARP) and Reverse Address Resolution Protocols (RARP) used in Transport Layer.

10. Study of Network Simulator-NS2.

11. Simulate any Dynamic Routing Protocol used to route the packets in Network Layer.

12. **Minor Project:**

   a) Design a simple textual chat application that resembles Talk command in UNIX.

   b) Implementation of ping server and client application using sockets.

REFERENCE BOOKS:

1. Andrew S Tanenbaum, David J. Wetherall, "*Computer Networks*," Pearson Education, 5th Edition, 2011.

2. Forouzan Behrouz A and MosharrafFirouz, "*Computer Networks A Top-Down Approach*," Tata McGraw Hill publications, 4th Edition, 2007.

3. http://www.ns2blogger.in.

(19MC30132) DATA WAREHOUSING AND DATA MINING LAB

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	3	1.5

PRE-REQUISITES:

A course on Data Warehousing and Data Mining.

COURSE DESCRIPTION:

Hands-on experience on developing active/passive transformations; Creation of Datasets; Data Preprocessing; Association Rule Mining; Classification and Clustering techniques using Data Warehouse ETL and WEKA tool.

COURSE OUTCOMES:
After successful completion of the course, students will be able to:

CO1: Design and develop solutions for data acquisition process from one data source to other target data source using ETL tool.

CO2: Select and Apply Classification and Clustering algorithms on preprocessed datasets to infer predictions effectively.

CO3: Apply methods of data mining to assess and provide analytical solutions for societal issues.

CO4: Analyze preprocessing techniques, data mining algorithms and identify frequent itemsets using Apriori algorithm to extract interesting patterns from large databases using WEKA components.

CO5: Function effectively as an individual and as a member in a team to manage and implement data mining applications**.**

CO6: Write and present a technical report/ document effectively.

LIST OF EXERCISES:

PART –A

Creation of Active/Passive transformations using Data Warehouse (Extract, Transform, Load) ETL Tool

1. Construct data acquisition process to extract, transform and load data from different databases.

2. Design and implement data acquisition process to perform

a) Expression Transformation

b) Joiner Transformation

3. Design and implement data acquisition process to perform

   a) Aggregator Transformation

   b) Source Qualifier Transformation

4. Design and implement data acquisition process to perform

   a) Filter Transformation

   b) Router Transformation

5. Design and implement data acquisition process to perform

   a) Ranker Transformation

   b) Sorter Transformation

PART -B

Working with Data Mining - WEKA tool.

6. Create data sets in ARFF and CSV formats and load into the Weka Explorer.

7. Perform data preprocessing steps on weather nominal and student information data sets as follows:

   a) Handle missing values for categorical and nominal values.

   b) Select relevant attributes.

   c) Apply normalization techniques

8. Generate strong Association rules by applying Apriori algorithm on preprocessed dataset with Min_Sup=60% and Min_Conf=80%.

9. i) Implement the Classification using Decision Tree algorithm on 'Weather' dataset. Draw the confusion matrix and report the model with accuracy.
   (ii) Implement Bayesian Classification and analyze the results on 'iris'
   dataset.

10. Implement simple k-Means clustering algorithm on 'iris' dataset.

11. Use Experimenter WEKA component to rank the performance of j48, oneR, ID3, algorithms on 'weather nominal' dataset.

12. Verify ID3 classifier performance using Gain ration and Ranker method using a Knowledge flow WEKA component.

13. Minor Project

Step 1: Creation of data set.

Step 2: Apply preprocessing techniques on constructed data sets.

Step 3: Implement appropriate data mining algorithms such as:

a. Apriori algorithm – to find frequent itemsets using various support and confidence levels

b. FP growth association mining

c. ID3 decision tree classifier

d. Build a confusion matrix to compute sensitivity, specificity, precision, recall, weighted accuracy and correlation between the attributes.

Data sets: Super Market data, Health data, Banking system, Weather forecasting and social media data.

REFERENCE BOOKS:

1. Ian H. Witten, Eibe Frank, and Mark,"*A Data Mining: Practical Machine Learning Tools and Techniques,"* Hall Morgan Kaufmann, 3rd Edition, 2011.

2. Ralph Kimball, *"The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling,"* John Wiley and Sons Inc, 3rd Edition, 2013.

3. G. K. Gupta, *"Introduction to Data Mining with Case Studies,"* PHI, New Delhi, 3rd Edition, 2009.

ADDITIONAL LEARNING RESOURCES:

1. Tutorial on Data Mining Algorithms by Ian.

2. *Mining of Massive Datasets* by Anand Rajaraman and Jeff Ullman.

3. Open source tools for data mining: http://eprints.fri.uni-lj.si/893/1/2008-OpenSourceDataMining.pdf

4. https://www.kaggle.com/datasets

MCA III – Semester

16MC30132: DATA WAREHOUSING AND DATA MINING LAB

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	3	2

PREREQUISITES: A course on "Data Warehousing and Data Mining".

COURSE DESCRIPTION:

Develop Transformations using Data Warehouse ETL tool; Creation of Datasets; Data Preprocessing; Association Rule Mining; Classification and Clustering; Multimedia, Text, Web Data Mining and Applications.

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

CO1. Demonstrate knowledge on:

- Data acquisition process

- Data preprocessing methods

- Data Mining algorithms

CO2. Analyze frequent itemsets using Apriori and FP-growth algorithms.

CO3. Design and construct data acquisition process from one data source to other target data source using data warehouse ETL tool.

CO4. Develop solutions for complex computing problems by applying appropriate   data mining algorithms to evaluate the accuracy and error measures using WEKA components.

CO5. Use WEKA tool to preprocess weather, hospital, and banking datasets to discover knowledge for making future predictions effectively.

CO6. Communicate effectively in implementing data mining problems with respect to documentation and visualization of hidden patterns.

CO7. Apply the knowledge of data mining to assess and provide computing solutions for societal issues.

CO8. Function effectively as an individual and as a member in a team to manage and implement data mining application in multidisciplinary environment.

LIST OF EXERCISES

PART –A

Creation of Active/Passive transformations using Data Warehouse (Extract, Transform, Load) ETL Tool

14. Construct data acquisition process to extract, transform and load data from different databases.

15. Design and implement data acquisition process to perform

    c) Expression Transformation

    d) Joiner Transformation

16. Design and implement data acquisition process to perform

    c) Aggregator Transformation

    d) Source Qualifier Transformation

17. Design and implement data acquisition process to perform

    c) Filter Transformation

    d) Router Transformation

18. Design and implement data acquisition process to perform

    c) Ranker Transformation

    d) Sorter Transformation

PART -B

Working with Data Mining - WEKA tool.

19. Creation on weather nominal and student results data sets in .arff and .csv formats

20. Perform data preprocessing steps on weather nominal and student information data sets as follows:

    d) Handling of missing values for categorical and nominal values.

    e) Selection of relevant attributes.

    f) Applying normalization techniques

21. Perform Association rule mining algorithm on preprocessed data set.

22. Perform classification and prediction on processed data set using J48 and ID3 algorithms.

23. Use Experimenter WEKA component to evaluate the accuracy and error measures of a classifier or predictor.

24. Verify ID3 classifier performance using Gain ration and Ranker method using a Knowledge flow WEKA component.

25. Minor Project

Step 1: Creation of data set.

Step 2: Apply preprocessing techniques on constructed data sets.

Step 3: Implement appropriate data mining algorithms such as:

    e. Apriori algorithm – to find frequent itemsets using various support and confidence levels

    f. FP growth association mining

    g. ID3 decision tree classifier

    h. Build a confusion matrix to compute sensitivity, specificity, precision, recall, weighted accuracy and correlation between the attributes.

Data sets: Super Market data, Health data, Banking system, Weather forecasting, social media and Iris data.

REFERENCE BOOKS:

4. Ian H. Witten, Eibe Frank, and Mark,"*A Data Mining: Practical Machine Learning Tools and Techniques,*" Hall Morgan Kaufmann, 3rd Edition, 2011.

5. Ralph Kimball, "*The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling,*" John Wiley & Sons Inc, 3rd Edition, 2013.

6. G. K. Gupta, "*Introduction to Data Mining with Case Studies,*" PHI, New Delhi, 3rd Edition, 2009.

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	3	1.5

PRE-REQUISITES:

A course on Python Programming.

COURSE DESCRIPTION:

Hands on practice–Scripting using Python Programming constructs; Conditional statements; Loops; Text Files; Lists; Dictionaries; Strings; Functions; GUI.

COURSE OUTCOMES:

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

CO1: Understand python programming constructs, data types, control structures, expressions, objects, lists, dictionaries, tuples, strings and text files to solve computational problems.

CO2: Identify and analyse classes, functions, modules, algorithms to solve problems.

CO3: Design and develop GUI based application using tkinter, Python programming and object-oriented programming constructs for Horse Race Simulation application.

CO4: Select and Apply python 3.7.5, Anaconda Navigator, Jupyter notebook to solve problems.

CO5: Work independently or in teams to solve problems with effective Communication.

CO6: Write and present a technical report/ document effectively.

LIST OF EXERCISES:

1. a. Write a python script to display a simple message.

    b. Write a python script to perform basic arithmetic operations on two values which are accepted from the user.

2. a. Write a python script to calculate the factorial of a given number.

    b. Write a python script to calculate sum of individual digits of a given number.

    c. Write a python script to display the prime number series up to the given N Value.

3. a.   Write a python script to find the largest number among three numbers and   display them in ascending order using if-else construct.

   b.   Write a python script to create a simple text file, write the contents into the created file and display the same on to the console screen.

4.   Write a python script to remove all the occurrences of a given character from a text file, copy the resultant text into another text file. Find the total occurrences of the eliminated characters and display the count along with the contents of the text file on to the console.

5. a.   Write a python script to display Fibonacci sequence of numbers using while loop and for loop constructs.

   b.   Write a python script to demonstrate string methods.

6. a.   Write a python script to create a list and add n number of user-defined values to the list and display the same on to the console screen.

   b.   Write a python script to perform the following operations on Lists:

        (i) Matrix Addition.

        (ii) Matrix Multiplication.

7. a.   Write a python script to search a key element in the given list of elements.

   b.   Write a python script to arrange the given list of elements in ascending or descending order.

8. a.   Write a python script to find GCD of two numbers using recursive and non recursive functions.

   b.   Write a python script to convert the following using functions:

        (i) Fahrenheit to Celsius temperature.

        (ii) Celsius to Fahrenheit temperature.

9. a.   Write a python script to draw a square using set position method in absolute positioning.

   b.   Write a python script to draw a triangle using left, right and Forward methods in relative positioning.

c. Write a python script using penup and pendown methods to draw "W" character using turtle graphics.

d. Write a python script to create your own polygon shape and create an interesting design with it.

10. a. Write a GUI Script for creating text label in a window.

b. Write a Python Script to create a command button. When the button is clicked the event should be handled and the message on the window should change from "Hello" to "Good Bye".

11. a. Write a python script to demonstrate the Exception Handling.

b. Write a Python script to demonstrate the Mouse and Key Event handling.

c. Write a python script to demonstrate menu driven applications

12. Form a group of 3 to 4 members develop a mini project for Horse Race simulation with the help of GUI programming and tkinter package.

REFERENCE BOOKS:

1. Charles Dierbach, "*Introduction to Computer Science using Python: A Computational Problem-Solving Focus,*" Wiley India Edition, 2016.

2. Mark Lutz, "*Programming Python,*" O'Reilly Publications, 4th Edition, 2011.

SOFTWARE / TOOLS USED:

Jupyter notebook, Tkinter, python 3.7.5, anaconda navigator

ADDITIONAL LEARNING RESOURCES:

1. https://docs.python.org/3/tutorial/

2. https://pythonprogramming.net/introduction-to-python-programming/

(19MC40101) OBJECT ORIENTED ANALYSIS AND DESIGN (Theory)

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		3	1	-	4

PRE-REQUISITES:

Courses on Object Oriented Programming through JAVA and Software Engineering.

COURSE DESCRIPTION:

Things and Classes; Relationships; Class Diagrams; Object Diagrams; Usecase Diagrams; Interaction Diagrams; Activity Diagrams; State Chart Diagrams; Component Diagrams; Deployment Diagrams.

COURSE OUTCOMES:

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

CO1: Analyze the specifications of Class, Things, principles of Object-Oriented development, Use case, Activity, Sequence and State diagrams to develop static and dynamic models using pre conditions and post conditions.

CO2: Design application artifacts to construct the Logical, Behavioral and Architectural models of an application using common modeling techniques of things.

CO3: Use UML tool Rational Rose or Visual Paradigm to design Class, Use Case, Sequence, Collaboration, Activity, State Chart, Component and Deployment Diagrams.

CO4: Function effectively as a member or leader in teams to analyze and design Leave Management System.

DETAILED SYLLABUS:

UNIT I- INTRODUCTION TO UML **(9 Periods)**

The meaning of Object Orientation, object identity, Importance of modeling, principles of modeling, object-oriented modeling, An overview of UML, conceptual model of the UML, Architecture.

Classes – Terms and concepts, Common Modeling Techniques.

Relationships – Modeling simple dependencies, single Inheritance and structural relationships, Common Mechanisms and UML Diagrams.

UNIT II- STRUCTURAL MODELING   (9 Periods)

Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages, Instances.

Class Diagrams – Terms, concepts, modeling techniques for Class Diagram, Modeling Simple collaboration, Logical database Schema, Forward and Reverse Engineering.

Object Diagrams – Modeling object structures, Forward and Reverse engineering.

UNIT III- BASIC BEHAVIORAL MODELING   (9 Periods)

Use cases – Terms and Concepts, Common Modeling techniques.

Use case Diagrams – Terms and Concepts, Common Modeling Techniques.

Sequence Diagrams – Terms and Concepts, Modeling flows of control by time ordering; **Collaboration Diagrams –** Terms and Concepts, Modeling flows of control by Organization, Forward and Reverse Engineering.

UNIT IV- ADVANCED BEHAVIORAL MODELING   (9 Periods)

Activity Diagrams – Terms and Concepts, Modeling a workflow, Modeling an operation, forward and reverse Engineering.

Events and Signals, State Machines, Processes and Threads, Time and Space, State Chart Diagrams – Modeling Reactive Objects.

UNIT V- ARCHITECTURAL MODELING   (9 Periods)

Component Diagrams – Terms and Concepts, Modeling Source Code, Modeling Physical Database, Forward and Reverse Engineering;

Deployment Diagrams – Terms and Concepts, Modeling Embedded System, Modeling Distributed System, Forward and Reverse Engineering.

Patterns and Frameworks, Artifact Diagrams.

Case study: Implementation of Component and Deployment diagrams using Rational Rose or Visual Paradigm for Leave Management System.

Total Periods: 45

Topics for self-study are provided in the lesson plan.

TEXT BOOK:

1. Grady Booch, James Ram Baugh and Ivar Jacobson, "*The Unified Modeling Language User Guide*," Pearson Education, 1999.

REFERENCE BOOKS:

1. John W. Satzinger, Robert B. Jackson and Stephen D. Burd, "*Object-Oriented Analysis and Design with the Unified Process,"* Cengage Learning, 2004.

2. Hans-Erik Eriksson, Magnus Penker, Brian Lyons and David Fado*, "UML 2: Toolkit,"* Wiley India Pvt. Ltd., 2004.

ADDITIONAL LEARNING RESOURCES:

1. https://nptel.ac.in/content/storage2/nptel_data3/html/mhrd/ict/text/106105153/
2. https://courses.cs.washington.edu/courses/cse403/11sp/lectures/lecture08-uml1.pdf
3. https://www.uml-diagrams.org/index-examples.html

16MC30103: OBJECT ORIENTED ANALYSIS AND DESIGN

Int. Marks	Ext.Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Object Oriented Programming Through JAVA" and "Software Engineering"

COURSE DESCRIPTION:

Things and Classes; Relationships; Class Diagrams; Object Diagrams; Usecase Diagrams; Interaction Diagrams; Activity Diagrams; State Chart Diagrams; Component Diagrams; Deployment Diagrams.

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

CO1. Demonstrate knowledge on Things, principles of Object Oriented Development.

CO2. Analyze the specifications of Class, Use case, Activity, Sequence and State diagrams and develop models using pre conditions and post conditions.

CO3. Design application artifacts to construct the Logical, Behavioral and Architectural model of an Application.

CO4. Solve complex behavior using common modeling techniques of things.

CO5. Make use of UML Tool such as Rational Rose or Visual Paradigm to design Class, Use Case, Sequence, Collaboration, Activity, State Chart, Component and Deployment Diagrams for the an Application.

DETAILED SYLLABUS:-

UNIT -I: INTRODUCTION TO UML   **(11 Periods)**

The meaning of Object Orientation, object identity, Importance of modeling, principles of modeling, object oriented modeling, An overview of UML, conceptual model of the UML, Architecture.

Classes - Terms and concepts, Common Modeling Techniques.

Relationships - Modeling simple dependencies, single Inheritance and structural relationships, Common Mechanisms and UML Diagrams.

UNIT-II: STRUCTURAL MODELING   **(12 Periods)**

Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages, Instances.

Class Diagrams - Terms, concepts, modeling techniques for Class Diagram, Modeling Simple collaboration, Logical database Schema, Forward and Reverse Engineering.

Object Diagrams – Modeling object structures, Forward and Reverse engineering.

UNIT-III: BASIC BEHAVIORAL MODELING   (11 Periods)

Use cases - Terms and Concepts, Common Modeling techniques.

Use case Diagrams - Terms and Concepts, Common Modeling Techniques.

Sequence Diagrams - Terms and Concepts, Modeling flows of control by time ordering; **Collaboration Diagrams –** Terms and Concepts, Modeling flows of control by Organization, Forward and Reverse Engineering.

UNIT-IV: ADVANCED BEHAVIORAL MODELING   (11 Periods)

Activity Diagrams - Terms and Concepts, Modeling a workflow, Modeling an operation, forward and reverse Engineering.

Events and Signals, State Machines, State Chart Diagrams – Modeling Reactive Objects.

UNIT-V: ARCHITECTURAL MODELING   (10 Periods)

Component Diagrams – Terms and Concepts, Modeling Source Code, Modeling Physical Database, Forward and Reverse Engineering;

Deployment Diagrams – Terms and Concepts, Modeling Embedded System, Modeling Distributed System, Forward and Reverse Engineering.

Case Study: The Unified Library Application.

Total Periods: 55

TEXT BOOK:

1. Grady Booch, James Ram Baugh and Ivar Jacobson, "*The Unified Modeling Language User Guide*," Pearson Education, 1999.

REFERENCE BOOKS:

1. John W. Satzinger, Robert B Jackson and Stephen D Burd, "*Object-Oriented Analysis and Design with the Unified Process,"* Cengage Learning, 2004.

2. Hans-Erik Eriksson, Magnus Penker, Brian Lyons and David Fado, "UML 2: Toolkit," Wiley India Pvt. Ltd., 2004.

(Professional Elective-I)

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PRE-REQUISITES:

Courses on Data Structures and Computer Networks.

COURSE DESCRIPTION:

Distributed Systems, Blockchain, Types of blockchains, Decentralization, Bitcoin, Alternative Coins, Smart Contracts, Ethereum 101, Applications of Ethereum, Scalability Challenges in Privacy and Security, Current Landscape.

COURSE OUTCOMES:

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

CO1: Understand the concepts of distributed Systems, decentralization and blockchain technology for the development of an application.

CO2: Select and use the Bitcoins, Smart Contracts and Ethereum 101 for the development of distributed systems and decentralized systems.

CO3: Analyze the issues and challenges of scalability, privacy and security in monetizing the businesses using blockchain technology.

CO4: Develop and deploy the applications using BlockApps, Eris platforms.

CO5: Commit to ethics and cyber regulations to perform encryption, multiparty computation, smart governance using Smart contract security.

DETAILED SYLLABUS:

UNIT I– DISTRIBUTED SYSTEMS AND DECENTRALIZATION   (12 Periods)

Distributed Systems: CAP Theorem, Byzantine Generals Problem, Consensus, History of blockchain, Generic Elements, Features, Applications, Tiers, Types of blockchain, CAP Theorem and blockchain, Benefits and limitations of blockchain.

Decentralization: Decentralization using blockchain, Methods of decentralization, Routes to Decentralization, blockchain and full ecosystem decentralization, Smart contract, Decentralized organizations, Decentralized autonomous organizations, Decentralized autonomous corporations, Decentralized autonomous societies, Decentralized applications, Platforms for decentralization.

UNIT II – DIGITAL CURRENCY **(11 Periods)**
Bitcoin: Concepts, Transactions, Blockchain, Bitcoin Payments.

Alternative Coins: Theoretical foundations, Bitcoin limitations, Namecoin, Litecoin, Primecoin, Zcash.

UNIT III – SMART CONTRACTS AND ALTERNATIVE BLOCKCHAINS **(10 Periods)**
Smart Contracts: History, Ricardian contracts, Smart Contract Templates, Oracles, Smart Oracles, Deployment of Smart Contracts on Blockchains.

Alternative Blockchains: Kadena; Platforms: BlockApps, Eris

UNIT IV – ETHEREUM 101 **(11 Periods)**

Ethereum: Ethereum blockchain, Elements of Ethereum blockchain, Precompiled contracts, Accounts, Block, Ether, Messages, Mining, Clients and wallets, Ethereum network, Applications developed on Ethereum, Scalability and Security Issues.

UNIT V – CHALLENGES AND CURRENT LANDSCAPE **(11 Periods)**

Scalability and Other Challenges: **Scalability**: Block size increase, Block internal reduction, Invertible Bloom lookup tables, Sharding, State channels, Private blockchain, Proof of Stake. **Privacy**: Indistinguishability obfuscation, Homomorphic encryption, Zero knowledge proofs,

State channels, Secure multiparty computation, Usage of hardware to provide confidentiality, Coinjoin, Confidential transactions, MimbleWimble; **Security**: Smart contract security.

Current Landscape: Emerging Trends, Improvement proposals; Blockchain research: Smart contracts, Centralization issues, Limitations in cryptographic functions, Consensus Algorithms, Scalability, Code Obfuscation.

Total Periods: 55
Topics for self-study are provided in the lesson plan.
TEXT BOOK:
1. Imran Bashir, "*Mastering Blockchain,*" Packt Publishing Ltd., 2017.
REFERENCE BOOKS:
1. Arshdeep Bahga, Vijay Madisetti, "*Blockchain Applications: A Hands-On Approach,*" VPT Books, 2017.
2. Josh Thompsons, "*Blockchain: The Blockchain For Beginners Guide To Blockchain Technology and Leveraging Blockchain Programming,*" Create Space Independent Publishing Platform, 2017.
ADDITIONAL LEARNING RESOURCES:
1. https://nptel.ac.in/courses/106105184/
2. https://medium.com/moatcoin/part-6-blockchain-simplified-notes-nptel-892f13875555
3. http://www.hands-on-books-series.com/assets/Bahga-Madisetti-Blockchain-Book-Code.zip

MCA IV - SEMESTER

(19MC40105) MULTIMEDIA APPLICATION DEVELOPMENT (Theory)

(Professional Elective-I)

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PRE-REQUISITES:

A course on Web Programming.

COURSE DESCRIPTION:

Multimedia Concepts; Data Representation; Action script Programming Concepts; Dynamic Action Script and Event Handling Mechanism; Video and audio compression Techniques and Multimedia communication and data transmission.

COURSE OUTCOMES:

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

CO1: Understand the concepts of Multimedia, hypermedia, validation of forms and action script programming and Event Handling.

CO2: Design and Develop online media applications using searching of motor vectors and Moving Picture Expert Group (MPEG) techniques for visual design.

CO3: Demonstrate Bitmap Programming, multimedia software tools, graphic software, scanning and digital photography to create original images.

CO4: Analyze image data representation graphics, image data types, file formats, color models in images and color models in videos to design graphics and videos.

CO5: Solve the compression of audio and video problems using audio and video compression techniques.

CO6: Apply compression standards, Adaptive Differential Pulse Code Modulations (ADPCM), Vocoders and Linear Predictive Coding (LPC) to develop quality applications.

DETAILED SYLLABUS:

UNIT I- MULTIMEDIA AUTHORING AND DATA REPRESENTATIONS   (11 Periods)

Multimedia Authoring and Data Representations: Multimedia and hypermedia, World Wide Web, overview of multimedia software tools. Graphics and image data representation graphics/image data types, file formats, Color in image and video: color science, color models in images, color models in video.

UNIT II- ACTION SCRIPT   (12 Periods)

64

UNIT III- DISPLAY AND INTERACTIVITY   (12 Periods)

Display and Interactivity: Events and Event Handling, Exceptions and Error Handling**,** Dynamic Action Script, Scope**,** Events and Display Hierarchies, Interactivity. Screen Updates, Programmatic Animation, Drawing with Vectors, Bitmap Programming, Text Display and Input.

UNIT IV- BASIC VIDEO COMPRESSION TECHNIQUES   (10 Periods)

Basic Video Compression Techniques: Introduction to video compression, video compression based on motion compensation, search for motion vectors, MPEG video coding I- MPEG-1 and 2, Basic Audio Compression Techniques: ADPCM in Speech Coding, G.726 ADPCM, Vocoders, LPC.

                                                              Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. Ze-Nian Li, and Mark S. Drew*, "Fundamentals of Multimedia,"* Pearson Education,2008.
2. Colin Moock, SPD O, REILLY*, "Essentials ActionScript 3.0,"* 2007.

REFERENCE BOOKS:

1. Nigel chapman, and jenny chapman, "*Digital Multimedia,"* Wiley-Dreamtech, 2005.
2. Fred Halsall, "*Multimedia Communications: Applications, Networks, Protocols and Standards,"* Pearson Education, 2001.

16MC50111: MULTIMEDIA AND RICH INTERNET APPLICATION DEVELOPMENT

(Professional Elective – IV)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Web Programming".

COURSE DESCRIPTION:

Concepts of Multimedia; Multimedia authoring tools; Data representations; Fundamental concepts in Video and digital audio; Basic video compression techniques; Multimedia communication and retrieval; Development of rich internet applications with adobe flash.

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

CO1.   Demonstrate Knowledge on multimedia, video compression techniques and adobe flash.

CO2.   Analyze a variety of creative techniques like sequential, hierarchical search and MPEG in the visual design of online media.

CO3.   Design and development of Multimedia Animations using Adobe Flash and Flex3

CO4.   Create highly interactive, rich internet applications using multimedia technologies and authoring tools.

CO5.   Evaluate the role and importance of critical experimentation and innovation in the multimedia development process as a professional practice.

DETAILED SYLLABUS:

UNIT- I: MULTIMEDIA AUTHORING AND DATA REPRESENTATIONS

(10 periods)

Multimedia and hypermedia, World Wide Web, overview of multimedia software tools. Graphics and image data representation graphics/image data types, file formats, Color in image and video: color science, color models in images, color models in video.

UNIT- II: FUNDAMENTAL CONCEPTS IN VIDEO AND DIGITAL AUDIO

(11 periods)

Types of video signals, analog video, digital video, digitization of sound, MIDI, quantization and transmission of audio. Multimedia Data Compression: Lossless compression algorithms, Lossy compression algorithms, Image compression standards.

UNIT-III: BASIC VIDEO COMPRESSION TECHNIQUES   **(11 periods)**

Introduction to video compression, video compression based on motion compensation, search for motion vectors, MPEG video coding I- MPEG-1 and 2, Basic Audio Compression Techniques: ADPCM in Speech Coding, G.726 ADPCM, Vocoders.

UNIT-IV: MULTIMEDIA COMMUNICATION AND RETRIEVAL   (12 periods)

Basics of Computer and Multimedia Networks, Multiplexing Technologies, LAN and WAN, Access Networks. Multimedia Network Communications and Applications: Quality of Multimedia Data Transmission, Multimedia over IP, Multimedia over ATM Networks, Transport of MPEG-4, Media-On- Demand (MOD).

UNIT-V: RICH INTERNET APPLICATIONS (RIAS) WITH ADOBE FLASH (11 periods)

Adobe Flash Introduction, Flash Movie Development, Learning Flash with Hands-on Examples, Publish your flash movie, creating special effects with Flash, Creating a website splash screen, action script. Rich Internet Applications (RIAs) with Flex3 Introduction, Flex Platform Overview, Creating a Simple user Interface.

Total Periods: 55

TEXT BOOKS:

1. Ze-Nian Li, and Mark S. Drew, "*Fundamentals of Multimedia*", Pearson Education, 2008.

2. Paul J Deitel and Harvey M Deitel, "*AJAX Rich Internet Applications, and Web Development for Programmers*", Deitel Developer Series, Pearson Education, 2009.

REFERENCE BOOKS:

1. Nigel Chapman, and jenny chapman, "*Digital Multimedia"*, Wiley-*Dreamtech, 2005*

2. Russel Chun, "*Flash CS3 Professional Advanced*", Pearson Education, 2007.

(19MC40106) **INFORMATION SECURITY (Theory)**

(Professional Elective – I)

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PRE-REQUISITES:

A course on Computer Networks.

COURSE DESCRIPTION:

Cryptographic algorithms; Classical Encryption Techniques; Public key and Private key encryption; Security models; Hash Algorithms; E-mail, IP and Web Security; ensuring system security and security over the Internet; Intrusion Detection and Trusted systems.

COURSE OUTCOMES:

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

CO1: Understand Network security model and cryptographic techniques for secure communication.
CO2: Apply Cryptographic techniques to provide security for E-Mail and IP based communication
CO3: Analyze Cryptographic algorithms and provide solutions for secure data transmission.
CO4: Demonstrate web and system security techniques to prevent the data from digital attacks.

DETAILED SYLLABUS:

UNIT I- COMPUTER AND NETWORK SECURITY CONCEPTS AND CLASSIC ENCRYPTION TECHNIQUES **(10 Periods)**

Computer Security Concepts, OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, A Model for Network Security, Standards.

Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques– Caesar Cipher, Monoalphabetic Ciphers, Playfair Cipher, Polyalphabetic Ciphers and Transposition Techniques.

UNIT II- CONFIDENTIALITY USING SYMMETRIC AND ASYMMETRIC CIPHERS

(12 Periods)

Block Ciphers and the Data Encryption Standard: Traditional Block Cipher Structure, Data Encryption Standard (DES), Strength of DES, Block Cipher Design Principles.

Advanced Encryption Standard and Block Cipher Operation: AES Structure, Multiple Encryption and Triple DES, Electronic Codebook, Cipher Block Chaining Mode, Cipher Feedback Mode, Output Feedback Mode and Counter Mode. Stream Ciphers and RC4, Placement of Encryption Function, Traffic Confidentiality.

Public-Key Cryptography: Principles of Public-Key Cryptosystems, Public-Key Cryptographic algorithms - The RSA Algorithm, Diffie – Hellman Key Exchange.

Case Study: Implement RSA and Diffie – Hellman Key Exchange algorithms.

UNIT III- CRYPTOGRAPHIC DATA INTEGRITY ALGORITHMS AND MUTUAL TRUST

(14 Periods)

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining and Secure Hash Algorithm (SHA).

Message Authentication Codes: Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MACs, MACs Based on Hash Functions: HMAC and MACs Based on Block Ciphers: DAA and CMAC.

Digital Signatures: Digital Signatures, NIST Digital Signature Algorithm.

Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates and Public-Key Infrastructure.

User Authentication: Remote User – Authentication Principles, Kerberos, Federated Identity Management, Personal Identity Verification.

Case Study: To check the integrity of files in a system using any open source security algorithm.

UNIT IV- ELECTRONIC MAIL SECURITY AND IP SECURITY   **(09 Periods)**

Electronic Mail Security: Email Threats and Comprehensive Email Security, S/MIME and Pretty Good Privacy (PGP).

IP Security: IP Security Overview, IP Security Policy, Authentication Header (AH), Encapsulating Security Payload (ESP), Combining Security Associations and Internet Key Exchange.

UNIT V- WEB SECURITY AND SYSTEM SECURITY   **(10 Periods)**

Transport-Level Security: Web Security Considerations, Secure Socket Layer and Transport Layer Security, Secure Electronic Transaction.

System Security: Intruders, Intrusion Detection systems, Viruses and Related Threats, Virus Countermeasures, Firewall Design Principles, Trusted Systems.

<mark>**Case Study:** A study on Challenges to mitigate security risks associated with Online and Mobile Payments</mark>

Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. William Stallings, "*Cryptography and Network Security (Principles and Practice),*" Pearson Education, 7th Edition, 2020.
2. William Stallings, "*Cryptography and Network Security,*" Pearson Education, 4th Edition, 2009.

REFERENCE BOOKS:

1. William Stallings, "*Network Security Essentials (Applications and Standards),*" Pearson Education, 3rd Edition, 2009.
2. Bernard L. Menezes, Ravinder Kumar, "*Cryptography, Network Security and Cyber Laws,*" Cengage Learning, 2019.
3. Behrouz A. Forouzan, "*Cryptography and Network Security,*" Tata McGraw-Hill, 2007.

ADDITIONAL LEARNING RESOURCES:

1. https://nptel.ac.in/courses/106105031/ lecture by Dr. Debdeep Mukhopadhyay, IIT Kharagpur
2. https://www.udemy.com/introduction-to-cryptography-online-course-rahsoft-crypto-certificate/ by Dr. Sourabh Prakash Head of Rahsoft Cyber Security and Cryprography Department
3. https://www.coursera.org/learn/asymmetric-cryptography offered by university of Colorado
4. https://www.khanacademy.org/computing/computer-science/cryptography-Journey into Cryptography.

16MC50106: INFORMATION SECURITY

(Professional Elective – III)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Computer Networks".

COURSE DESCRIPTION:

cryptographic algorithms; Classical Encryption Techniques; Public key and Private key encryption; security models; Hash Algorithms; E-mail and IP Security; analysis of security principles in internet and system security; Intrusion Detection.

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

CO1.   Demonstrate knowledge on

- Symmetric and Asymmetric Encryption Algorithms

- Key distribution and message Authentication

- Hash algorithms and digital signature techniques

- IP security and Web Security

- Intrusion Detection and Firewall configurations

CO2.   Analyze appropriate Symmetric, Asymmetric Encryption algorithms and Hash Algorithms to provide Confidentiality and Authentication.

CO3.   Design solutions to problems related to Public-Key Encryption, Digital signatures, Secure Hash Functions.

CO4.   Identify efficient ciphers such as Gauss Cipher, Vigenere cipher, Rail Fence Cipher and cryptographic algorithms such as RSA, Diffie-Hellman cryptographic algorithms, Digital Signature standard for Hashing techniques to provide novel solutions for real-time application protocols like PGP, S/MIME, SSL, TLS and SET.

CO5.   Use the Cryptographic Techniques - Vigenere cipher, Rail Fence Cipher to provide confidentiality, security Algorithms and hashing techniques to enhance level of protection in area of digital communication.

CO6.   Commit to ethics in authentication and access control methods to implement policies and mechanisms on business operations using Digital Signature Standards.

DETAILED SYLLABUS:

UNIT-I: INTRODUCTION TO SECURITY, CLASSIC ENCRYPTION TECHNIQUES

(11 Periods)

Introduction: Introduction to Security - Security Trends, The OSI Security Architecture, Security Attacks, Security Services and Mechanisms. A model for Network security, Internet Standards and the Internet Society.

Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques – Caesar Cipher, Monoalphabetic Ciphers, Playfair Cipher, Hill Cipher, Polyalphabetic Ciphers and One – Time pad. Transposition Techniques.

Block Ciphers and the Data Encryption Standard: Block Cipher Principles, The Data Encryption Standard, The Strength of DES, Multiple Encryption and Triple DES.

UNIT-II: CONFIDENTIALITY USING CONVENTIONAL ENCRYPTION TECHNIQUES

(11 Periods)

Advanced Encryption Standard: The AES Cipher, Block Cipher Modes of Operation, Stream Ciphers and RC4, Placement of Encryption Function, Traffic Confidentiality, Key Distribution.

Public-Key Cryptography: Principles of Public-Key Cryptosystems, Public-Key Cryptography algorithms - The RSA Public-Key Encryption Algorithm, Diffie – Hellman Key Exchange, Key Management.

UNIT-III: MESSAGE AUTHENTICATION, HASH FUNCTIONS AND DIGITAL SIGNATURE STANDARD   (12 Periods)

Message Authentication: Authentication Requirements, Authentication Functions, Message Authentication Codes.

Hash Functions: Hash Functions, Secure Hash Algorithm – SHA-512 Logic and Round Function, HMAC.

Digital Signatures and Authentication Protocols: Digital Signatures, Authentication Protocols, Digital Signature Standard.

Authentication Applications: Kerberos, X.509 Authentication Service.

UNIT-IV: ELECTRONIC MAIL SECURITY AND IP SECURITY   (9 Periods)

Electronic Mail Security: Pretty Good Privacy (PGP) and S/MIME.

IP Security: IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.

UNIT-V: WEB SECURITY AND SYSTEM SECURITY (12 Periods)

Web Security: Web Security Considerations, Secure Socket Layer and Transport Layer Security, Secure Electronic Transaction.

System Security: Intruders, Intrusion Detection systems, Viruses and Related Threats, Virus Countermeasures, Firewall Design Principles, Trusted Systems.

Case Study: To check the integrity of files in a system using any open source security algorithm.

Total Periods: 55

TEXT BOOK:

1. William Stallings, "*Cryptography and Network Security*", Pearson Education,

   4th Edition, 2009.

REFERENCE BOOKS:

1. William Stallings, "*Network Security Essentials (Applications and Standards)*", Pearson Education, 3rd Edition, 2009.

2. Behrouz A. Forouzan, "*Cryptography and Network Security*", Tata McGraw-Hill, 2007.

3. Charlie Kaufman, Radia Perlman and Mike Speciner, "*Network Security - Private Communication in a Public World*", Pearson Education, 2nd Edition, 2005.

4. Michael E. Whitman, Herbert J. Mattord, "*Principles of Information Security*", Cengage Learning, 2008.

MCA IV - SEMESTER

(Professional Elective - II)

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PRE-REQUISITES:

Courses on Mathematical Foundations of Computer Science and Data Structures.

COURSE DESCRIPTION:

AI Problems; Problem Characteristics Search Algorithms; Inference in Propositional Logic; axioms of probability; baye's rule; decision theory; computational learning theories; Basic Probability Notations; Forms of Learning; fuzzy logic.

COURSE OUTCOMES:

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

CO1: Understand the concepts of Artificial Intelligence, Problem characteristics, problem solving agents, production system and knowledge representation to solve AI problems.

CO2: Formulate Logical Agents, First-Order Logic and Inference in First-Order Logic to compute intelligent systems.

CO3: Apply knowledge representation using probabilistic and statistical reasoning, propositional theorems, reasoning theories and models to recommend advisory consultative situation.

CO4: Analyse artificial Intelligent Techniques, searching algorithms, strategies, problems, Propositional and first order Logic to infer solutions for Grammar Induction and implementation methodology problem.

CO5: Use statistical methods, distributions, genetic algorithms, learning and evolutionary strategies to optimize particle swarns problems.

DETAILED SYLLABUS:

UNIT I - ARTIFICIAL INTELLIGENCE   **(11 Periods)**

AI Problems, Underlying Assumption, Levels of the Model, Criteria of Success, Some General References.

Problems, Problem Spaces, and Search: Define the Problem as a State Space Search, Production Systems, Problem Characteristics, Production System Characteristics.

Use Case: Solve Water Jug problem using state space and production rules.

UNIT II - SEARCHING AND PROBLEM SOLVING   (12 Periods)

Solving Problems by Searching: issues, problem solving agents, searching for solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions,

Beyond classical search: Local Search Algorithms and Optimization Problems, Local Search in Continuous Spaces, searching with nondeterministic actions, searching with partial observations.

UNIT III - KNOWLEDGE AND REASONING   (11 Periods)

Logical Agents: Knowledge Based Agents, Wumpus World, Logic, Propositional Logic, Propositional theorem Proving, Effective Propositional model checking, Agents Based on Propositional Logic.

First-Order Logic: Representation Revisited, Syntax and Semantic of First-Order Logic, Using First-Order Logic, Knowledge Engineering in First-Order Logic.

Inference in First-Order Logic: Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution.

UNIT IV - KNOWLEDGE REPRESENTATION AND REASONING   (12 Periods)

Ontological Engineering, Categories and Objects, Events, Mental Events and Mental Objects, Reasoning Systems for Categories, Reasoning with Default Information, Internet Shopping World.

Uncertainty Knowledge and reasoning: Acting under Uncertainty, Basic Probability Notation, the Axioms of Probability, Inference Using Full Joint distributions, Independence, Bayes' Rule and its use.

Probabilistic reasoning: Representing knowledge in uncertain domain, hidden Markov model.

Statistical Reasoning: certainty factors and rule-based systems, Dampster-shafer theory, fuzzy logic.

Case Study: Sales advisory-consultative situation in buying a complex technical product.

UNIT V - LEARNING AND EVOLUTIONARY COMPUTATION CONCEPTS

(11 Periods)

Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. Elaine Rich, Kevin Knight and Shivashankar B Nair, "*Artificial Intelligence,*" Tata McGraw Hill, 3rd Edition, 2007.
2. Stuart Russell and Peter Norvig, "*Artificial Intelligence A Modern Approach*," Pearson Education, 2nd Edition, 2011.

REFERENCE BOOKS:

1. Russell C.Eberhart and Yuhui Shi, "*Computational Intelligence: Concepts to Implementations,*" Elsevier, 2007.
2. George F. Luther, "*Artificial Intelligence: Structures and Strategies for Complex Problem Solving,*" Pearson Education, 5th Edition, 2001.

ADDITIONAL LEARNING RESOURCES:

1. https://ai.google/education/
2. https://nptel.ac.in/courses/106/105/106105078/
3. https://www.slideshare.net/ravislides99/problems-problem-spaces-and-search-112060118
4. https://www.researchgate.net/publication/262250593_Genetic_algorithms_Concepts_issues_and_a_case_study_of_grammar_induction
5. https://cseweb.ucsd.edu/classes/sp07/cse150/lectures-pdf/l.newsearch.pdf
6. https://towardsdatascience.com/basic-probability-theory-and-statistics-3105ab637213

(Professional Elective-II)

Int. Marks	Ext. Marks	Total Marks		L	T	P	C
40	60	100		4	-	-	4

PRE-REQUISITES:

A course on Computer Networks.

COURSE DESCRIPTION:

Computer Security and threats, Browser Attacks, Security in the design of Operating Systems, Wireless Network Security, Intrusion Detection and Prevention Systems, Privacy Impacts of Emerging Technologies, Managing the incidents.

COURSE OUTCOMES:

After successful completion of this course, the student will be able to:

CO1: Understand the concepts of Threats, Harm, Vulnerabilities, Security in Operating Systems, Security Counter measures, Internet of things, Data Mining, Big Data and privacy concepts for providing Cyber Security.

CO2: Identify incidents of Economics, Electronic Voting and Cyber Warfare and privacy impacts of emerging techniques to identify the critical issues.

CO3: Apply security counter measures, intrusion detection and prevention techniques to provide Network Security.

CO4: Commit ethics of cyber regulations, responsibilities, and norms to manage incidents using privacy principles, policies, Cyber Welfare and International Laws to adapt in cyberspace.

DETAILED SYLLABUS:

UNIT I– CYBER SECURITY   (9 periods)

Computer Security, Threats, Harm, Vulnerabilities, Controls, Authentication, Access Control and Cryptography, Web User Side, Browser Attacks, Web attacks Targeting Users, Obtaining User or Website Data, Email Attacks.

UNIT II – SECURITY IN OPERATING SYSTEM AND NETWORKS   (11 periods)
Security in Operating Systems, Security in the Design of Operating Systems, Rootkit, Network security attack, Threats to Network Communications, Wireless Network Security, Denial of Service, Distributed Denial-of-Service.

Cryptography in Network Security, Firewalls, Intrusion Detection and Prevention Systems, Network Management, Databases, Security Requirements of Databases, Reliability and Integrity, Database Disclosure, Data Mining and Big Data.

UNIT IV – PRIVACY IN CYBERSPACE   **(12 periods)**

Privacy Concepts, Privacy Principles and Policies, Authentication and Privacy, Data Mining, Privacy on the Web, Email Security, Privacy Impacts of Emerging Technologies, Where the Field Is Headed.

UNIT V – MANAGEMENT AND INCIDENTS   **(11 periods)**

Security Planning, Business Continuity Planning, Handling Incidents, Risk Analysis, Dealing with Disaster, Emerging Technologies, The Internet of Things, Economics, Electronic Voting, Cyber Warfare, Cyberspace and the Law, International Laws, Cyber crime, Cyber Warfare and Home Land Security.

Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOK:

1. Charles P. Pfleeger Shari Lawrence Pfleeger Jonathan Margulies, *"Security in Computing,"* 5th Edition, Pearson Education, 2015.

REFERENCE BOOKS:

1. George K.Kostopoulous, *"Cyber Space and Cyber Security,"* CRC Press, 2013.
2. MarttiLehto, PekkaNeittaanmäki, *"Cyber Security: Analytics, Technology and Automation"*, Springer International Publishing Switzerland, 2015.

ADDITIONAL LEARNING RESOURCES:

1. https://www.edx.org/course/cyber,security,basics,a,hands,on,approach.
2. https://www.cyberaces.org/courses/
3. https://www.futurelearn.com/courses/introduction,to,cyber,security
4. https://swayam.gov.in/nd2_cec20_cs09/preview

(19MC40111) SERVICE ORIENTED ARCHITECTURE (Theory)

(Professional Elective-II)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PRE-REQUISITES:

Courses on Software Engineering, Computer Networks and Web Programming.

COURSE DESCRIPTION:

XML document structure; Web Services; Principles of SOA; Service Layers; Simple Object Access Protocol (SOAP); Web Services Description Language (WSDL) and Building SOA-Based Applications.

COURSE OUTCOMES:

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

CO1: Understand the concepts of XML, Characteristics of SOA, Benefits of SOA, Principles of Service orientation and Service layers.

CO2: Design and develop modern softwares using XML parsers, WSDL and SOAP Web services.

CO3: Analyze XML Transformation, XSL Formatting, Service Oriented Analysis and Design, Service Modeling and guidelines to build XML and SOA based applications.

CO4: Commit to SOA standards to design service oriented web applications for an individual and society.

DETAILED SYLLABUS:

UNIT I- INTRODUCTION TO XML   **(11 periods)**

XML document structure, well formed and valid documents, Namespaces, DTD, XML Schema, X-Files.

UNIT II- BUILDING XML- BASED APPLICATIONS   **(11 periods)**

Parsing XML: using DOM, SAX, XML Transformation and XSL, XSL Formatting, Modeling databases in XML.

UNIT III- SERVICE ORIENTED ARCHITECTURE   **(11 periods)**

Characteristics of SOA, Comparing SOA with Client/Server and Distributed architectures, Benefits of SOA , Principles of Service orientation and Service layers.

UNIT IV- WEB SERVICES   **(11 periods)**

Service descriptions, WSDL, Messaging with SOAP, Service discovery, UDDI, Message Exchange Patterns, Orchestration, Choreography and WS Transactions.

Total Periods: 55

Topics for self-study are provided in the lesson plan.

TEXT BOOKS:

1. Thomas Erl**, "***Service-Oriented Architecture: Concepts, Technology and Design,"* Pearson Education, 2013. ISBN 978-81-317-1490-4.
2. Ron Schmelzer et al. *"XML and Web Services,"* Pearson Education, 2002.

REFERENCE BOOKS:

1. Frank P. Coyle, *"XML, Web Services and the Data Revolution,"* Pearson Education, 2002.
2. Eric Newcomer, Greg Lomow, *"Understanding SOA with Web Services,"* Pearson Education, 2005
3. Sandeep Chatterjee and James Webber, *"Developing Enterprise Web Services: An Architect's Guide,"* Prentice Hall, 2004.
4. James McGovern, Sameer Tyagi, Michael E.Stevens, Sunil Mathew*, "Java Web Services Architecture,"* Morgan Kaufmann Publishers, 2003.