DEPARTMENT OF INFORMATION TECHNOLOGY
B.Tech-R21 COURSE STRUCTURE

III B.Tech I SEMESTER

S.No	Course Code	Course Title	Category	L	T	P	C
1	21IT3111	Design and Analysis of Algorithms	PC	3	-	-	3
2	21CS3112	Software Engineering	PC	3	-	-	3
3	21CS3113	Web Technologies	PC	3	-	-	3
4	**Open Elective-I**		OE	3	-	-	3
5	**Professional Elective-I**		PE	3	-	-	3
	21IT3171	Embedded Systems					
	21CS3171	Data Analytics					
	21IT3172	Advance Operating Systems					
	21IT3173	Software Project Management					
6	21IT3151	Software Engineering and R Programming Lab	PC	-	-	3	1.5
7	21CS3152	Web Technologies Lab	PC	-	-	3	1.5
8	21HS3153	Advanced English Communication Skills Lab	HS	-	-	2	1
9	21MC0006	Aptitude And Logical Reasoning	MC	3	-	-	0
10	21IT3181	Summer Internship	PW	-	-	-	1
11	21MC0003	Environmental Science	MC	2	-	-	0
Total			**28**	**20**	**-**	**8**	**20**

Note: Summer Internship carried out during summer break after II Year II semester.

III B.Tech II Semester

S.No	Course Code	Course Title	Category	L	T	P	C
1	21CS3211	Cryptography and Network Security	PC	3	1	-	3
2	21IT3211	Compiler Design	PC	3	-	-	3
3	21CS3213	Machine Learning	PC	3	-	-	3
4	**Open Elective-II**		OE	3	-	-	3
5	**Professional Elective-II**		PE	3	-	-	3
	21IT3271	Cloud Computing					
	21IT3272	Mobile Computing					
	21IT3273	Software Testing Methodologies					
	21CS3274	Distributed Databases					
6	21IT3251	Cryptography and Network Security Lab	PC	-	-	2	1.5
7	21IT3252	Compiler Design Lab	PC	-	-	3	1.5
8	21CS3253	Machine Learning Lab	PC	-	-	3	1.5
9	21MC0005	Indian Constitution	MC	3	-	-	0
10	21MC0007	Yoga & Indian Philosophy	MC	3	-	-	0
Total			**27**	**18**	**1**	**8**	**19.5**

21IT3111: DESIGN AND ANALYSIS OF ALGORITHMS

III B. Tech I Sem. **L T P C**
3 - - 3

Pre- requisites:
- Knowledge on basic mathematics and programming concepts

Course Objectives:
- To analyze performance of algorithms.
- To understand and choose the appropriate algorithm design technique for a specified application.
- To solve problems using algorithm design techniques such as the greedy method, divide and. conquer, dynamic programming, backtracking and branch and bound.
- To analyze the impact of algorithm design techniques on each application solved.
- To introduce and understand P and NP classes.

Course Outcomes:
- Able to analyze the different algorithm design techniques for a given problem.
- Able to design algorithms for various computing problems.
- Able to argue the correctness of algorithms using inductive proofs and invariants.
- Able to synthesize set operations.
- Able to explain about coping with the limitations of algorithms.

UNIT I

Notation of an Algorithm: Fundamentals of Algorithmic Problem Solving, Fundamentals of the Analysis of Algorithm Efficiency–Order Notations and its properties, Mathematical analysis for Recursive -Towers of Hanoi and Non-recursive algorithms, Randomized Algorithms-Monte Carlo and Las Vegas, Amortized analysis Examples.

Divide and conquer: General method-Control abstraction, Solving Recurrence Relation using Substitution method and Master's Theorem, applications - Binary search, Merge sort, Quick sort, Strassen's Matrix Multiplication, Finding Maximum and Minimum element.

UNIT II

Disjoint Set Operations: Union and find algorithms, AND/OR graphs, Graph traversals-Breadth first search, Depth First search, Connected Components and Spanning trees, Bi-connected components.

Greedy method- General method-Control abstraction, applications- Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Single source shortest path problem.

UNIT III

Dynamic Programming: General Method, applications-Multi Stage Graphs, All pairs shortest path problem, Optimal binary search trees, 0/1 knapsack problem, Reliability design, Traveling sales person problem.

UNIT IV

Backtracking: General method-Control abstraction, applications-The 8-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.
Branch and Bound: General Method-Control abstraction, applications-15-Puzzle Problem-LC search, 0/1 Knapsack problem-LC Branch and Bound solution, FIFO Branch and Bound solution, Traveling sales person problem.

UNIT V
NP-Hard and NP-Complete problems: Basic concepts, Non-deterministic algorithms, NP – Hard and NP- Complete classes, Cook's theorem- proof of reduction.
Approximation Algorithms for NP – Hard Problems: Traveling Salesman problem, Knapsack problem.

Text Books:

1. Ellis Horowitz, SatrajSahni and S Rajasekharam, Fundamentals of Computer Algorithms, Galgotia publishers.
2. M.T. Goodrich, Robert Tamassia, Algorithm design: Foundations, Analysis and Internet examples, Wiley student Edn, John Wiley & sons.
3. Parag Himanshu Dave, Himanshu Bhalchandra Dave, Design and Analysis algorithms Pearson Publication.

Reference books:

1. Allen Weiss, Data structures and Algorithm Analysis in C++, 2nd Edn, Pearson Education Thomas H.Cormen, Charles
2. E.Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Third Edition, PHI Learning Private Limited.

21CS3112: SOFTWARE ENGINEERING

III B.Tech I Sem. **L T P C**

3 - - 3

Prerequisites:

- Computer Programming
- Database Management Systems

Course Objectives:

- To understand fundamental principles of Software engineering, and their application in the development of software products.
- To understand and create the software requirements specifications document.
- To understand and use unified modeling language for specifying, analysis and designing.
- To understand testing strategies for testing software applications
- To understand Software metrics, Risk Management strategies and software process improvement.

Course Outcomes

- Able to apply the software engineering lifecycle phases communication, planning, analysis, design, construction, and deployment.
- Ability to translate end-user requirements into system and software requirements into Software Requirements specification Document (SRS)
- Able to apply UML in object-oriented software modeling to develop computer software.
- Able to identify problems in software and will be able to develop a simple testing report.
- Able to apply Software Metrics to mitigate risks thereby improving software process.

UNIT I

Introduction: The evolving role of software, Changing Nature of Software, legacy software, Software myths.

A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models,

Process Models: The waterfall model, Incremental process models, Evolutionary process models, specialized process models, The Unified process,

Agile development- Agile process, Extreme Programming, scrum, dynamic systems development method, agile model, Agile Unified Process

UNIT II

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management, Software requirements documents.

System models: Context Models, Behavioral models, Data models, Object models, structured methods.

UNIT III

Modeling Techniques using UML: The Unified Approach to Modeling, Structural and Behavioral Diagrams.
Design Engineering: Data Flow Diagrams, Design process and Design quality, Design concepts, the design model, pattern-based software design.
Creating an architectural design: Architectural styles and patterns, Architectural Design, assessing alternative architectural designs.

UNIT IV
Implementation: Structured coding Techniques, Coding Styles-Standards and Guidelines.
Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Regression Testing, Unit Testing, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.
Quality Management: Quality concepts, software quality assurance, software reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

UNIT V
Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.
Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan.
Software Process Improvement: What is SPI, SPI Process, SPI Frameworks, Emerging trends in software engineering?

Text Books:
1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 8th edition, Mc Graw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

References:
1. "Software Engineering", Ian Sommerville, Addison-Wesley, 9th Edition, 2010, ISBN- 13:978-0137035151.
2. Richard Fairley, "Software Engineering Concepts", Tata McGraw Hill.
3. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.
4. Software testing techniques by Boris Beizer, dreamtech.
5. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, JohnWiley.
6. Software Engineering principles and practice- Waman S Jawadekar, The Mc Graw-Hill Companies.

21CS3113: WEB TECHNOLOGIES

III B.Tech I Sem. **LT P C**

3 - - 3

Course Objectives:

- To introduce Client-side scripting with java script and ajax
- To introduce PHP language for server-side scripting
- To introduce XML and processing of XML Data with java
- To introduce Server-side programming with java servlets
- To introduce server-side programming with JSP

Course Outcomes:

- Understand basics of HTML AND CSS and Design and Development of Dynamic Web Pages (using Validations) with Java Script and AJAX programming.
- Understand Server-side Scripting with PHP language.
- Analyze what is XML and how to parse and use XML data with JAVA.
- Develop Server side Applications with Servlets (Sessions and Cookies).
- Create JSP pages with Database Server.

UNIT I

HTML Common tags- List, Tables, images, forms, Frames; Cascading Style sheets;
Client-side Scripting: Introduction to JavaScript, JavaScript language – declaring variables, scope of variables, functions, event handlers (onclick, on submit etc.), Document Object Model, Form validation.

UNIT II

XML: XML: Introduction to XML, XML document structure, DTD, Namespaces and XML Schemas.
Web Application Frameworks: AngularJS, ReactJS, NodeJS, JQuery.

UNIT III

Introduction to Servlets: Common Gateway Interface (CGI), Life cycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Session tracking, Cookies and Sessions, connecting to a database using JDBC.

UNIT IV

Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP.

Unit V

Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, and lists etc., Handling File Uploads. Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies
File Handling in PHP: File operations - opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

Text Books:

1. Web Technologies, Uttam K Roy, Oxford University Press
2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill

References:

1. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley DreamTech.
2. Java Server Pages –Hans Bergsten, SPD O'Reilly
3. Java Script, D. Flanagan, O'Reilly, SPD.
4. Beginning Web Programming-Jon Duckett WROX.
5. Programming World Wide Web, R. W. Sebesta, Fourth Edition, Pearson.
6. Internet and World Wide Web – How to program, Dietel and Nieto, Pearson

21IT3171: EMBEDDED SYSTEMS
(Professional Elective-I)

III B.Tech I Sem. **L T P C**

3 - - 3

Prerequisites

- A course on "Digital Logic Design".
- A course on "Computer Organization".

Course Objectives:

- Introduce the basic concepts of an embedded system.
- Explain various elements of embedded hardware and their design principles
- Elaborate different steps involved in the design and development of firmware for embedded systems.
- Discuss Internals of Real-Time operating system, the fundamentals of RTOS based embedded firmware design and fundamental issues in hardware software co-design.
- Familiarize with different embedded system implementation and testing tools.

Course Outcomes:

- Explain the basic concepts and the embedded system design approach to perform a specific function.
- Analyze the hardware components required for an embedded system and the design approach of an embedded hardware.
- Analyze various embedded firmware design approaches on embedded environment.
- Evaluate the issues in hardware software co-design.
- Integrate hardware and firmware of an embedded system using real time operating systems.

UNIT I

Embedded system-Definition, History of embedded systems, classification of embedded systems, major application areas of embedded systems, purpose of embedded systems, the typical embedded system-core of the embedded system, Memory, Sensors and Actuators, Communication Interface, Embedded firmware, Characteristics of an embedded system, Quality attributes of embedded systems, Application-specific and Domain-Specific examples of an embedded system.

UNIT II

Embedded hardware design: Analog and digital electronic components, I/O types and examples, Serial communication devices, Parallel device ports, Wireless devices, Timer and counting devices, Watchdog timer, Real time clock.

UNIT III

Embedded firmware design: Embedded Firmware design approaches, Embedded Firmware development languages, ISR concept, Interrupt sources, Interrupt servicing mechanism, Multiple interrupts, DMA, Device driver programming, Concepts of C versus Embedded C and Compiler versus Cross-compiler.

UNIT IV

Real time operating system: Operating system basics, Types of operating systems, Tasks, Process and Threads, Multiprocessing and Multitasking, Task Scheduling, Threads, processes and Scheduling, Task communication, Task synchronization.

Hardware software co-design: Fundamental Issues in Hardware Software Co-Design, Computational models in embedded design, Hardware software Trade-offs, Integration of Hardware and Firmware.

UNIT V

Embedded system development, implementation and testing: The integrated development environment, Types of files generated on cross-compilation, Dissembler/DE compiler, Simulators, Emulators and Debugging, Target hardware debugging, Embedded Software development process and tools, Interpreters, Compilers and Linkers, Debugging tools, Quality assurance and testing of the design, Testing on host machine, Simulators, Laboratory Tools.

Text Books:

1. Embedded Systems Architecture-By Tammy Noergaard, Elsevier Publications, 2013.
2. Embedded Systems-By Shibu. K.V-Tata McGraw Hill Education Private Limited, 2013.

References:

1. Embedded System Design, Frank Vahid, Tony Givargis, John Wiley Publications, 2013.
2. Embedded Systems-Lyla B.Das-Pearson Publications, 2013.

21CS3171: DATA ANALYTICS

(Professional Elective - I)

B.Tech. III Year I Sem. **L T P C**

3 - - 3

Prerequisites

- Data Base Management Systems, Computer Oriented Statistical Methods

Course Objectives

- To explore the fundamental concepts of data analytics.
- To learn the principles and methods of statistical analysis.
- To develop problem solving abilities using Mathematics.
- To apply algorithmic strategies while solving problems.
- To understand the various search methods and visualization techniques.

Course Outcomes

- Identify the various sources of Big Data.
- Apply several key big data technologies used for storage, analysis and manipulation of data.
- Design methodologies to extract data from structured and un-structured data for analytics
- Apply Regression algorithms to perform data analytics.
- To carry out standard data visualization and formal inference processors.

UNIT I

Data Management: Design Data Architecture and manage the data for analysis, understand various sources of Data like Sensors/Signals/GPS etc. Data Management, Data pre-processing, Missing Values - Outlier Detection and Treatment.

UNIT II

Introduction to Tools and Environment, Application of Modelling in Business, Databases & Types of data and variables, Data Modelling Techniques, Missing imputations etc. Need for Business Modeling.

UNIT III

Regression – Concepts, Blue property assumptions, Least Square Estimation, Variable Rationalization, and Model Building etc.

Logistic Regression: Model Theory, Model fit Statistics, Model Construction, Analytics applications to various Business Domains etc.

UNIT IV

Supervised and Unsupervised Learning, Tree Building – Regression, Classification, Over fitting, Pruning and Complexity, Multiple Decision Trees etc.

Time Series Methods: Arima, Measures of Forecast Accuracy, STL approach, Extract features from generated model as Height, Average Energy etc and Analyze for prediction.

UNIT V

Data Visualization: Pixel Oriented Visualization Techniques, Geometric Projection Visualization Techniques, Icon based Visualization Techniques Hierarchical Visualization Techniques, Visualizing Complex Data and Relations.

Text Books:

1. Data Mining Concepts and Techniques, Han, Kamber, 3rd Edition, Morgan Kaufmann Publishers.
2. Associate Analytics Handbook.

References:

1. Mining of Massive Datasets, Jure Leskovec Stanford Univ. Anand Rajaraman Milliway Labs Jeffrey D Ullman Stanford Univ.
2. Michael Minelli, Michele Chambers, AmbigaDhiraj ,"Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends", John Wiley & Sons, 2013.
3. Bart Baesens, "Analytics in a Big Data World: The Essential Guide to Data Science and its Applications", John Wiley & Sons, 2014

21IT3172: ADVANCED OPERATING SYSTEMS
(Professional Elective-I)

III B.Tech I Sem. **L T P C**

3 - - 3

Course Objectives:

- Understand concepts of Unix file operating system and memory management.
- Understand the concepts of processes and its relationship.
- To understand the concepts of file system and its types.
- To know the security issues and protection mechanisms for distributed environments.
- Understand the concepts of protection principles and security tools.

Course outcomes:

- Understand the Basic approaches of advanced operating systems.
- Analyze the Processor and its relationships.
- Evaluate file systems and system calls.
- Identify the requirements Distributed File System and Distributed Shared Memory.
- Gain Knowledge on protection and security of the system.

UNIT I

Introduction: Basic Operating System Concepts, Overview of Unix File System, Files, Links, Types, Inodes, Access Rights, System Calls, Overview of Unix Kernels, Model, Implementation, Reentrant Kernels, Address Space, Synchronization and Critical region, Inter process Communication, Process Management, Memory Management, Device Drivers.

UNIT II

Processes: Processes, Lightweight Processes, and Threads, Process Descriptor, State Identifying a Process, Relationships among Processes, Organization - Resource Limits, Creating Processes, System Calls, Kernel Threads, Destroying Processes, Termination, Removal, Interruption and Exceptions

UNIT III

File system: The Virtual File System (VFS),Role , File Model , System Calls, Data Structures, Super Block, Inode, File, Dentry Objects, Dentry Cache, Files Associated with a Process, File system Types, Special File systems, File system Type Registration, File system Handling, Namespaces, Mounting, Un mounting, Implementation of VFS System Calls.

UNIT IV

Distributed system: Advantages of Distributed system, Types of network-based operating system, Network structure, Communication structure, Communication protocols.

Distributed shared memory-Architecture: Algorithms for implementing DSM, Memory Coherence and Protocols, Design Issues

UNIT V

Protection and security: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix.

The Security Problem: Program threats, System and network threats, Cryptography as a security tool,

User authentication, Implementing security defences, Firewalling to protect systems and networks

Text Books:

1. Mukesh Singhal, Niranjan G.Shivaratri, "Advanced concepts in operating systems: Distributed, Database and multiprocessor operating systems", TMH, 2001
2. Wiseman, Yair: Advanced Operating Systems and Kernel Applications, Techniques and Technologies. Information Science Publishing, 2009

References:

1. Andrew S.Tanenbaum, "Modern operating system", PHI, 2003
2. Pradeep K.Sinha, "Distributed operating system-Concepts and design", PHI, 2003
3. Russinovich, Solomon: Windows Internals, 5th edition. Microsoft Press, 2009

21IT3173: SOFTWARE PROJECT MANAGEMENT
(Professional Elective-I)

III B. Tech I Sem. **L T P C**

3 - - 3

Prerequisites:

- A Knowledge on Software Engineering

Course Objectives:

- To acquire knowledge on software process management
- To acquire managerial skills for software project development
- To understand software economics

Course Outcomes:

- Gain knowledge of software economics, phases in the life cycle of software development,
- project organization, project control and process instrumentation
- Analyze the major and minor milestones, artifacts and metrics from management and technical
- perspective
- Design and develop software product using conventional and modern principles of software project management

UNIT I

Software Process Maturity: Software maturity Framework, Principles of Software Process Change, Software Process Assessment, The Initial Process, The Repeatable Process, The Defined Process, The Managed Process, The Optimizing Process**.**

Process Reference Models: Capability Maturity Model (CMM), CMMI, PCMM, PSP, TSP).

UNIT II

Software Project Management Renaissance: Conventional Software Management, Evolution of Software Economics, Improving Software Economics, The old way and the new way.

Life-Cycle Phases and Process artifacts: Engineering and Production stages, inception phase, elaboration phase, construction phase, transition phase, artifact sets, management artifacts, engineering artifacts and pragmatic artifacts, model-based software architectures.

UNIT III

Workflows and Checkpoints of process: Software process workflows, Iteration workflows, Major milestones, minor milestones, periodic status assessments.

Process Planning: Work breakdown structures, Planning guidelines, cost and schedule estimating process, iteration planning process, Pragmatic planning.

UNIT IV

Project Organizations: Line-of- business organizations, project organizations, evolution of organizations, process automation**.**

Project Control and process instrumentation: The seven-core metrics, management indicators, quality indicators, life-cycle expectations, Pragmatic software metrics, metrics automation.

UNIT V

CCPDS-R Case Study and Future Software Project Management Practices

Modern Project Profiles, Next-Generation software Economics, Modern Process Transitions.

Text Books:

1. Managing the Software Process, Watts S. Humphrey, Pearson Education
2. Software Project Management, Walker Royce, Pearson Education

References:

1. An Introduction to the Team Software Process, Watts S. Humphrey, Pearson Education, 2000
2. Process Improvement essentials, James R. Persse, O'Reilly, 2006
3. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, TMH, 2006
4. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly, 2006.
5. Head First PMP, Jennifer Greene & Andrew Stellman, O'Reilly, 2007
6. Software Engineering Project Management, Richard H. Thayer & Edward Yourdon, 2nd edition, Wiley India, 2004.
7. Agile Project Management, Jim Highsmith, Pearson education, 2004.

21IT3151: SOFTWARE ENGINEERING AND R PROGRAMMING LAB

III B. Tech I Sem. **L T P C**

0 0 3 1.5

Prerequisites:

- Basic Knowledge on Computer programming
- Knowledge on Database Management systems

Course Objectives:

- To understand fundamental principles of Software engineering, and their application in the development of software products.
- To understand and create the software requirements specifications document.
- To understand and use unified modelling language for specifying, analysis and designing.
- To understand testing strategies for testing software applications
- To understand Software metrics and Risk Management strategies to identify potential problems before they occur.
- Effective use of Business Intelligence (BI) technology (Tableau) to apply data visualization
- To discern patterns and relationships in the data.

Course Outcomes:

- Able to recognize the software engineering lifecycle phases.
- Ability to determine end-user requirements and software requirements and translate them into Software Requirements specification Document (SRS)
- Able to select an appropriate architectural model with design engineering.
- Able to assess problems in software and to write a simple testing report.
- Able to determine Software Metrics, potential risk and how to manage them though RMMM plan.
- Understand Tableau concepts of Dimensions and Measures.
- Develop Programs and understand how to map Visual Layouts and Graphical Properties.

List of Experiments for Software Engineering:

Do the following 6 exercises for any two projects given in the list of sample projects or any other projects using smart draw, Rational Rose or Star UML for UML diagrams:

1) Development of problem statement.
2) Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3) Preparation of Software Configuration Management and Risk Management related documents.
4) Draw level 0, level 1, and level 2 dataflow diagrams
5) Study and usage of any Design phase CASE tool .
6) Performing the Design by using any Design phase CASE tools.
7) Develop test cases for unit testing and integration testing.
8) Develop test cases for various white box and black box testing techniques.

Sample Projects:

1) Hospital management system
2) Online mobile recharge portal
3) Online Exam Registration
4) Stock Maintenance System
5) Online course reservation system
6) E-ticketing

List of Experiments for R Programming:

1. Write an R-Program to take input from user.
2. Write an R Program to Find the Fibonacci sequence Using Recursive Function
3. Write an R-Program to demonstrate working with operators.
4. Write an R Program to Check if a Number is Odd or Even
5. Write an R Program to check if the given Number is a Prime Number
6. Write an R Program to Find the Factorial of a Number
7. Write an R Program to Find L.C.M of two numbers
8. Write an R Program to create a Vector and to access elements in a Vector
9. Write an R Program to create a Matrix and access rows and columns using functions colnames() and rownames()
10. Write an R Program to create a List and modify its components.
11. Write an R Program to create a Data Frame.
12. Write an R Program to access a Data Frame like

    i) List ii) Matrix

Text Books:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, and Pearson Education.
4. Microsoft Power BI cookbook, Brett Powell, 2nd edition.
5. R Programming for Data Science by Roger D. Peng.

References:

1 "Software Engineering", Ian Sommerville, Addison-Wesley, 9th Edition, 2010, ISBN- 13: 978-0137035151.

2 Richard Fairley, "Software Engineering Concepts", Tata McGraw Hill.

3. The Art of R Programming by Norman Matloff Cengage Learning India

21CS3152: WEB TECHNOLOGIES LAB

III B. Tech I Sem. **L T P C**

- - 3 1.5

Course Objectives:

- To enable the student to program web applications using the following technologies HTML, JavaScript , AJAX, PHP, Tomcat Server, Servlets, JSP

Course Outcomes:

- USE LAMP / XAMP for Web Applications
- Simple Applications with Technologies like HTML, JavaScript and AJAX
- Design web application using PHP
- Parse XML Files using JAVA(DOM AND SAX Parsers)
- Use Tomcat Server for Servlets and connect to Database
- Develop JSP Applications using Tomcat Server and Java Bean development

EXPERIMENTS:

List of Experiments

1. Write an HTML code to demonstrate
   a) Lists b) Tables (row span and col span) c) Cascading Style Sheets

2. Design a web page to demonstrate
   a) Divisions b) Frames c) Embedding Images

3. Develop static pages (use Only HTML) of an online book store. The pages should resemble: www.amazon.com. The website should consist the following pages.
   a) Home page
   b) Registration and user Login
   c) User Profile Page
   d) Books catalog
   e) Shopping Cart
   f) Payment By credit card
   g) Order Conformation

4. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).

5. Write a JavaScript program to validate the registration form contents with the following Rules (Use RegExp Object)
   a) Username Must starts with Uppercase followed by set of lowercase letters or digits.
   b) Password must contain only uppercase letters and length must be in between 8 to12.
   c) Phone number contains 10 digits.
   d) E-mail must follow some predefined format (example@domain.com)

6. Build an application on a) React JS b) Angular JS c) Node.js and JSON.

7. Write the following XML Programs
   a) Create a DTD document to validate the XML document.
   b) Create a XML Schema document to validate the XML document

8. Install the following on the local machine
   a) Apache Tomcat Web Server
   b) Install MySQL/Oracle (if not installed)
   d) Install PHP and configure it to work with Apache web server and MySQL

9. a) Write a Servlet program to read the parameters from user interface and display Welcome message.
   b) Write a Servlet program to read initialization parameters using ServletConfig and Servlet Context object.

10. Write Servlet programs to work with the following session tracking techniques.
    a) Http Session b) Cookies c) Hidden form controls

11. Develop a dynamic web page which contains Registration and Login Forms using servlet with Oracle database .Validate the login page.

12. Write a JSP Program to handle the exceptions.

13. Develop a dynamic web page which contains Registration and Login Forms using JSP with Oracle database .Validate the login page.

14. Write a PHP script that reads data from one file and write into another file.

15. Develop a dynamic web page which contains Registration and Login Forms in PHP with MySQL database .Validate the login page.

Text Books:

1. WEB TECHNOLOGIES: A Computer Science Perspective, Jeffrey C. Jackson, Pearson Education

References:

1. Deitel H.M. and Deitel P.J., "Internet and World Wide Web How to program", Pearson International, 2012, 4th Edition.
2. J2EE: The complete Reference By James Keogh,McGraw-Hill
3. Bai and Ekedhi, The Web Warrior Guide to Web Programming,Thomson
4. Paul Dietel and Harvey Deitel," Java How to Program", Prentice Hall of India, 8thEdition
5. Web technologies, Black Book, Dreamtech press.

21HS3153: ADVANCED ENGLISH COMMUNICATION SKILLS LAB

III B. Tech I Sem. **L T P C**

- - 2 1

Introduction

A course on Advanced English Communication Skills (AECS) Lab is considered essential at the third year level of B.Tech and Pharmacy courses. At this stage, the students need to prepare themselves for their career which requires them to listen to, read, speak and write in English both for their professional and interpersonal communication. The main purpose of this course is to prepare the students of Engineering for their placements.

Course Objectives:

This Lab focuses on using multi-media instruction for language development to meet the following targets:

- To improve students' fluency in spoken English
- To enable them to listen to English spoken at normal conversational speed
- To help students develop their vocabulary
- To read and comprehend texts in different contexts
- To communicate their ideas relevantly and coherently in writing
- To make students industry-ready
- To help students acquire behavioural skills for their personal and professional life
- To respond appropriately in different socio-cultural and professional contexts

Course Outcomes: Students will be able to

- Acquire vocabulary and use it contextually
- Listen and speak effectively
- Develop proficiency in academic reading and writing
- Increase possibilities of job prospects
- Communicate confidently in formal and informal contexts

UNIT I

Inter-personal Communication and Building Vocabulary –Starting a Conversation–Responding Appropriately and Relevantly –Using Appropriate Body Language –Role Play in Different Situations – Synonyms and Antonyms, One-word Substitutes, Prefixes and Suffixes, Idioms and Phrases and Collocations.

UNIT II

Reading Comprehension –General Vs Local Comprehension, Reading for Facts, Guessing Meanings from Context, , Skimming, Scanning, Inferring Meaning

UNIT III

Writing Skills –Structure and Presentation of Different Types of Writing –Letter Writing/Resume Writing/ e-correspondence/ Technical Report Writing.

UNIT IV

Presentation Skills –Oral Presentations (individual or group) through JAM Sessions/Seminars/PPTs and Written Presentations through Posters/Projects/Reports/ emails/Assignments…etc.

UNIT-V

Group Discussion and Interview Skills –Dynamics of Group Discussion, Intervention, Summarizing, Modulation of Voice, Body Language, Relevance, Fluency and Organization of Ideas and Rubrics of Evaluation-Concept and Process, Pre-interview Planning, Opening Strategies, Answering Strategies, Interview through Tele-conference & Video-conference and Mock Interviews.

References:

1. Kumar, Sanjay and Pushp Lata. English for Effective Communication, Oxford University Press, 2015.
2. Konar, Nira. English Language Laboratories –A Comprehensive Manual, PHI Learning Pvt. Ltd., 2011

21MC0006: APTITUDE AND LOGICAL REASONING

III B. Tech I Sem. **L T P C**
3 - - 0

Course Objectives:

1.Student learns the techniques to solve all the problems in his real life.
2.It can improve the numerical ability.
3.The quicker methods are useful to solve the problems within the time and it is helpful in his duties.
4.Quantative Aptitude helps in solving the practical life problems.
5.Students can use Quantitative Aptitude in everyday life to figure out mathematically.
6.Student can improve his mental capacity.
7.It helps in sharpening their minds.

UNIT I

Number System, Percentages, Profit And Loss, Simple Interest - Compound Interest, Partnership Ratio And Proportion ,Chain Rule, Time And Work - Pipes And Cistern, Time And Distance - Problems On Trains, Boats And Streams, Races And Games Of Skill

UNIT II

Average, Alligation And Mixture, Permutation-Combination, Probability, Geometry (Co-Ordinate, Solid-2d Areas & 3d Volumes), D I (Tabulation, Bar Graphs, Pie Charts & Line Graphs), Elementary Statistics

UNIT III:

Series Completion, Analogy, Classification / Odd One Out, Coding – Decoding, Blood Relations, Deciphering Jumbled up Descriptions,

UNIT IV:

Relation Puzzle, Direction sense test, Number, Ranking & Time Sequence Test, Puzzle Test, Seating Arrangements Comparison Type Questions, Sequential Order of Things, Selection Based on gave conditions,

UNIT V

Family – Based Puzzles, Jumbled Problems. Logical Venn Diagrams
Alpha Numeric Sequence Puzzle, Cubes, Dice, Clocks, Calendar, Data Sufficiency, Syllogism.

TEXT BOOKS:

1. Quantitative Aptitude by R.S.Agarwal

2. Quantitative Aptitude by Abhijit Guha

3. Quantitative Aptitude for Competative Examinations, U.Mohan Rao, Scitech Publication.

21IT3181: SUMMER INTERNSHIP

III B. Tech I Sem. **L T P C**

- - - 1

21MC0003: ENVIRONMENTAL SCIENCE

III B. Tech I Sem. **L T P C**
2 - - 0

Course Objectives: To learn

- Identify the importance of ecosystem and its functions.
- Understand the natural resources and their usage in day to day life.
- Understand the concept of bio-diversity, its values and conservation.
- Be aware of the causes of different types of pollution and its control.
- Understand various environmental impacts, requirement of various policies and legislations towards environmental sustainability.

Course Outcomes: After the completion of the course, the student would be able to –

- Explain ecosystem and its functions namely, food chain, ecological pyramids etc.
- Acquire knowledge about different types of natural resources such as land, water, minerals, non-renewable energy and their excessive usage leading to detrimental effects on environment.
- Comprehend ecosystem diversity, its values and importance of hot spots to preserve the same.
- Explain different types of pollution, its control and impact on global environment.
- Recognize various environmental impacts and the importance of various acts and policies towards environmental sustainability.

UNIT I

Ecosystems: Definition, Scope, and Importance of ecosystem, Classification, structure, and function of an ecosystem, Food chains, food webs, and ecological pyramids, Flow of energy, Biogeochemical cycles, Bioaccumulation, Bio magnifications, Field visits.

UNIT II

Natural resources-Classification of Resources: Living and Non-Living resources,
Water resources: use and over utilization of surface and ground water, floods and droughts, dams benefits and problems, Environmental effects of extracting and using mineral resources,
Land resources: Forest resources, Energy Resources-renewable and non-renewable.

UNIT III

Biodiversity and biotic resources: Introduction, Definition, genetic, species and ecosystem diversity. Value of biodiversity; consumptive use, productive use, social, ethical, aesthetic and optional values.
Hot spots of biodiversity &Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflict, conservation of biodiversity: In-Situ and Ex-situ conservation. National Biodiversity act.

UNIT IV

Environmental pollution and control technologies: Environmental Pollution: Classification of pollution, Air Pollution: Primary and secondary pollutants, Automobile and Industrial pollution, Ambient air quality standards.
Water pollution: Sources and types of pollution, drinking water quality standards.
Soil Pollution: Sources and types, Impacts of modern agriculture, degradation of soil.
Noise Pollution: Sources and Health hazards, standards, Solid waste: Municipal Solid Waste management, composition and characteristics of e-Waste and its management. Pollution control technologies
Wastewater Treatment methods: Primary, secondary and Tertiary. Overview of air pollution control technologies.

Global Environmental Issues and Global Efforts: Green House Gases And its effect, Climate change and impacts on human environment. Ozone depletion and Ozone depleting substances (ODS). International conventions / Protocols: Earth summit, Kyoto protocol, and Montréal Protocol.

UNIT V

Environmental policy, legislation & eia: Environmental Protection act, Legal aspects Air Act- 1981, Water Act, Forest Act, Wild life Act, Municipal solid waste management and handling rules, hazardous waste management and handling rules.

EIA: EIA structure, methods of baseline data acquisition. Overview on Impacts of air, water, biological and Socio-economic aspects. Strategies for risk assessment, Concepts of Environmental Management Plan (EMP).

Towards Sustainable Future: Concept of Sustainable Development Goals, Population and its explosion, Crazy Consumerism, Environmental Education, Urban Sprawl, Human health, Environmental Ethics, Concept of Green Building, Ecological Foot Print, Life Cycle assessment (LCA), Low carbon life style.

Text Books:

1. ErachBharucha, Textbook of Environmental Studies for Undergraduate Courses, University Grants Commission.
2. R. Rajagopalan, Environmental Studies, Oxford University Press.

References:

1. Environmental Science: towards a sustainable future by Richard T. Wright. 2008 PHL Learning Private Ltd. New Delhi.
2. Environmental Engineering and science by Gilbert M. Masters and Wendell P. Ela. 2008 PHI Learning Pvt. Ltd.
3. Environmental Science by Daniel B. Botkin & Edward A. Keller, Wiley INDIA edition.
4. Environmental Studies by Anubha Kaushik, 4th Edition, New age international publishers.
5. Text book of Environmental Science and Technology - Dr. M. Anji Reddy 2007, BS Publications. 6. Introduction to Environmental Science by Y. Anjaneyulu, BS.Publications.

21CS3211: CRYPTOGRAPHY AND NETWORK SECURITY

III B. Tech II Sem. **L T P C**
3 1 - 3

Prerequisites:
- Should have knowledge of computer networks

Course Objectives
- To impart knowledge on network security issues, services, goals, and mechanisms.
- To classify various algorithms and fundamental ideas of public-key cryptography.
- To explain the authentication codes and digital signatures.
- To analyze the security of communication systems, networks, and protocols.
- To generate and distribute a PGP key Pair and use the PGP package to send an encrypted email message.

Course Outcomes
- Demonstrate knowledge of cryptography and network security concepts and applications
- Ability to compare various cryptographic algorithms
- Ability to identify the requirements and security of Message Authentication and hash functions
- Ability to apply security principles in networks and mobile devices
- Ability to analyze various E-Mail privacy techniques

UNIT I
Introduction to Cryptography: Introduction, The need for security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography Concepts and Techniques, symmetric and asymmetric key cryptography, steganography.

UNIT II
Symmetric and Asymmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, Block cipher modes of operation, Stream ciphers. Principles of public key cryptosystems, RSA algorithm, Elgamal Cryptography, Diffie-Hellman Key Exchange.

UNIT III
Message Authentication Algorithms and Hash Functions: Message Authentication, Secure Hash Algorithm (SHA512), Message authentication codes: Authentication requirements, authentication functions, message authentication code, hash functions, security of hash functions and MACS, Distribution of Public Keys, Kerberos, X.509 Authentication Service.

UNIT IV
Transport-level Security: Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS.
Wireless Network Security: Mobile Device Security, IEEE 802.11 Wireless LAN, IEEE 802.11i Wireless LAN Security
UNIT V
E-Mail and IP Security: Pretty Good Privacy, S/MIME , IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations.
Case Studies on Cryptography and security: Secure Multiparty Calculation, Virtual Elections.

Resources:
Video Lectures

1. http://nptel.ac.in/courses/106105031/lecture by Dr. Debdeep Mukhopadhyay IIT Kharagpur
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-033- computer-system-engineering-spring-2009/video-lectures/ lecture by Prof. Robert Morris and Prof. Samuel Madden MIT.

Text Books:

1. Cryptography and Network Security- Principles and Practice: William Stallings, Pearson Education, 6th Edition.
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition
3. Information Security, Principles, and Practice: Mark Stamp, Wiley India.

References:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, 3rd Edition
3. Wade Trappe, Lawrence C Washington, "Introduction to Cryptography with coding theory", Pearson.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning W. Mao,
7. "Modern Cryptography – Theory and Practice", Pearson Education.
8. Security in computing : Charles P. Pfleeger, Shari Lawrence Pfleeger

21IT3211: COMPILER DESIGN

III B. Tech II Sem. **L T P C**

3 - - 3

Prerequisites

- A course on "Formal Languages and Automata Theory"
- A course on "Computer Organization and architecture"
- A course on "Computer Programming and Data Structures"

Course Objectives

- To design, develop, and implement a compiler for any language.
- Understanding pre-processing steps required for Top-down and Bottom up parsing.
- To know what are Semantic analysis and Intermediate forms of source program and where are these applied.
- To design algorithms to generate machine code and to know how the data will be stored in Symbol tables i.e., storage allocation.
- To apply code optimization techniques in order to improve the performance of a program in terms of space and time complexity.

Course Outcomes

- An ability to design, develop, and implement a compiler for any language.
- Acquires a knowledge of pre-processing steps required for Top-down and Bottom up parsing.
- Acquire a fundamental understanding of what are Semantic analysis and Intermediate forms of source program and where are these applied.
- An ability to design a simple code generator and also acquire the knowledge of how the data will be stored in Symbol tables i.e, storage allocation
- Apply the theoretical concepts and techniques of code optimization in order to improve the performance of a program in terms of space and time complexity.

UNIT - I

Introduction: The structure of a compiler, the science of building a compiler, programming language basics, Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Specification and Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer Generator, Optimization of DFA.

UNIT - II

Syntax Analysis: Introduction, Context-Free Grammars, writing a Grammar, Left Recursion and Left Factoring, FIRST and FOLLOW, Top-Down Parsing: Recursive Descent Parsing, Predictive Parser, LL(1), Bottom-Up Parsing: Shift Reduce Parser and its conflicts, Operator Precedence Parsing Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using Ambiguous Grammars and Parser Generators, YACC.

UNIT - III

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's.

Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Type Checking, Control Flow, Switch-Statements, Intermediate Code for Procedures.

UNIT - IV

Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection, Introduction to Trace-Based Collection. **Code Generation:** Issues in the Design of a Code Generator, A Simple Code Generator, Register Allocation and Assignment, Dynamic Programming Code-Generation.

UNIT- V

Code Optimization: The Principal Sources of Optimization, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, Peephole Optimization, and Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, and Loops in Flow Graphs.

TEXT BOOKS:

1. Hopcroft, John E.; Motwani, Rajeev; Ullman, Jeffrey D. (2013). Introduction to Automata Theory, Languages, and Computation (3rd Ed.). Pearson. ISBN 1292039051
2. Kamala Krithivasan and Rama. R, "Introduction to Formal Languages, Automata Theory and Computation", Pearson Education 2009
3. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekaran, 2nd edition, PHI.

REFERENCES:

1. John C Martin, "Introduction to Languages and the Theory of Computation", Third Edition, Tata McGraw Hill Publishing Company, New Delhi, 2007
2. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning
3. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.

21CS3213: MACHINE LEARNING

III B.Tech II Sem. **L T P C**

3 - - 3

Prerequisites:

- Data Structures
- Knowledge on statistical methods

Course Objectives:

- Define Machine Learning and understand the basic theory underlying machine learning.
- Understand the basic concepts of learning and decision trees.
- Understand neural networks and Bayesian techniques for problems appear in machine learning.
- Understand the instance based learning and reinforced learning.
- Perform statistical analysis of machine learning techniques.

Course Outcomes: After the completion of the course student should be able to

- Choose the learning techniques and investigate concept learning
- Identify the characteristics of decision tree and solve problems associated with Decision tree Learning
- Apply effectively Neural Networks for appropriate applications
- Apply Bayesian techniques and derive effectively learning rules
- Evaluate hypothesis and investigate instance based learning, reinforced learning and Analytical Learning

UNIT I

Introduction - Well-posed learning problems, designing a learning system, Perspectives and issues in machine learning

Concept learning and the general to specific ordering – introduction, a concept learning task, concept learning as search, find-S: finding a maximally specific hypothesis, version spaces and the candidate elimination algorithm, remarks on version spaces and candidate elimination, inductive bias, Gradient Descent Algorithm and its variants.

UNIT II

Supervised Learning- Regression: Linear-Simple, Multiple, Logistic Regression.

Classification- Naive Bayes Classifier, k-NN classifier, Support Vector Machines -Linear, Non Linear

Ensemble Techniques I-Decision Trees-ID3(Iterative Dichotomiser3), CART(Classification and Regression Tree)

UNIT III

Ensemble Techniques II- C4.5, CHAID (Chi-Square Automatic Interaction Detection), Random Forest Algorithm.

Unsupervised Learning-Clustering: Measures of distance, k-means, Gaussian Mixture Model Clustering, Hierarchical Learning- Divisive, Agglomerative Clustering

UNIT IV

Artificial Neural Networks-1– Introduction, neural network representation, appropriate problems for neural network learning, perceptions, multilayer networks and the back-propagation algorithm.

Artificial Neural Networks-2- Remarks on the Back-Propagation algorithm, An illustrative example: face recognition, advanced topics in artificial neural networks.

UNIT V

Genetic Algorithms – Motivation, Genetic algorithms, an illustrative example, hypothesis space search, genetic programming, models of evolution and learning, parallelizing genetic algorithms.

Reinforcement Learning – Introduction, the learning task, Q–learning, non-deterministic, rewards and actions, temporal difference learning, generalizing from examples, relationship to dynamic programming.

Text Books:

1. Tom. M Mitchell, Machine Learning, McGraw Hill, 1997.
2. Trevor has tie, Robert Tibshirani & Jerome Friedman. The Elements of Statically Learning, Springer Verlag, 2001.
3. Introduction to Machine Learning with Python, Author – Andreas C. Müller, Sara h Guido, Edition – First Edition, Publisher – O'Reilly Media, Inc.

References:

1. Machine Learning Methods in the Environmental Sciences, Neural Networks, William W Hsieh, Cambridge UnivPress.
2. Richard o. Duda, Peter E. Hart and David G. Stork, pattern classification, John Wiley & SonsInc.,2001.
3. Chris Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.
4. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis.
5. Mathematics for Machine learning, Author – Marc Peter Deisenroth, Edition – First Edition, Publisher – Cambridge University Press.

21IT3271: CLOUD COMPUTING
(PROFESSIONAL ELECTIVE-II)

III B.Tech II Sem. **L T P C**

3 - - 3

Pre-requisites:

- A course on "Computer Networks".
- A course on "Operating Systems".

Course Objectives:

- To explain the evolving computer model called cloud computing.
- To introduce the various levels of services that can be achieved by cloud.
- To describe the security aspects in cloud.

Course Outcomes:

- Ability to define the different types of Computing models and their uses.
- Ability to explain characteristics of Cloud Computing.
- Ability to analyze and illustrate the cloud computing architecture.
- Ability to classify the types of Services and explain its advantages and disadvantages.
- Ability to summarize all Cloud service providers features.

UNIT I

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano Computing.

UNIT II

Cloud Computing Fundamentals: Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud Computing, Cloud Computing Is a Service, Cloud Computing Is a Platform, Principles of Cloud computing, Five Essential Characteristics, and Four Cloud Deployment Models.

UNIT III

Cloud Computing Architecture and Management: Cloud Architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications, on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure Managing the Cloud Application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

UNIT IV

Cloud Service Models: Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models.

UNIT V

Cloud Service Providers: EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue ,Service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by

SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform

Text Books:

1. Essentials of cloud Computing: K. Chandrashekaran, CRC press, 2014.

References:

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M.Goscinski, Wiley, 2011.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011

21IT3272: MOBILE COMPUTING
(PROFESSIONAL ELECTIVE-II)

III B. Tech II Sem. **L T P C**

3 - - 3

Prerequisites:

- Basic Knowledge of computer fundamentals
- Knowledge on computer networking

Course Objectives:

- To make the student understand the concept of mobile computing paradigm, its novel applications and limitations.
- To understand the typical mobile networking infrastructure through a popular GSM protocol
- To understand the issues and solutions of various layers of mobile networks, namely MAC layer, Network Layer & Transport Layer
- To understand the database issues in mobile environments & data delivery models.
- Understand the ad hoc networks, the platforms and protocols used in mobile environment.

Course Outcomes:

- Able to think and develop new mobile application.
- Able to take any new technical issue related to this new paradigm and come up with a solution(s).
- Solve the issues of all the layers of Mobile Networks.
- Able to develop new ad hoc network applications and/or algorithms/protocols.
- Able to understand & develop any existing or new protocol related to mobile. environment

UNIT I

Introduction to Mobile Communications and Computing: Mobile Computing (MC): Introduction to MC, novel applications, limitations, and architecture.
GSM: Mobile services, System architecture, Radio interface, Protocols, Localization and calling (Wireless) Medium Access Control: Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA.

UNIT II

Mobile Network Layer: Mobile IP (Goals, assumptions, entities and terminology, IP packet delivery, agent advertisement and discovery, registration, tunneling and encapsulation, optimizations), Dynamic Host Configuration Protocol (DHCP).

UNIT III

Mobile Transport Layer: Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery, Transmission /time-out freezing, Selective retransmission, Transaction oriented TCP.

UNIT IV

Data Dissemination: Communications asymmetry, classification of new data delivery mechanisms, push-based mechanisms, pull-based mechanisms, hybrid mechanisms, selective tuning (indexing) techniques.

UNIT V

Mobile Ad hoc Networks (MANETs): Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms, security in MANETs.
Protocols and Tools: Wireless Application Protocol-WAP. (Introduction, protocol architecture, and treatment of protocols of all layers), Bluetooth (User scenarios, physical layer, MAC layer, networking, security, link management) and J2ME

Text Books:

1. Jochen Schiller, "Mobile Communications", Addison-Wesley. (Chapters 4, 7, 9, 10, 11), second edition, 2004.
2. Stojmenovic and Cacute, "Handbook of Wireless Networks and Mobile Computing", Wiley, 2002, ISBN 0471419028. (Chapters 11, 15, 17, 26 and 27)

References:

1. Reza Behravanfar, "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", ISBN: 0521817331, Cambridge University Press, October 2004,
2. Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden, Schwiebert, Loren, "Fundamentals of Mobile and Pervasive Computing", ISBN: 0071412379, McGraw-Hill Professional, 2005.
3. Hansmann, Merk, Nicklous, Stober, "Principles of Mobile Computing", Springer, second edition, 2003.
4. Martyn Mallick, "Mobile and Wireless Design Essentials", Wiley DreamTech, 2003.

21IT3273: SOFTWARE TESTING METHODOLOGIES
(PROFESSIONAL ELECTIVE-II)

III B. Tech II Sem. **L T P C**

3 - - 3

Prerequisites:

- A course on "Software Engineering"

Course Objectives:

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in performing manual testing.
- To develop skills in software test automation and management using latest tools.
- To provide knowledge in test case planning for manual and automation testing.
- To provide knowledge in taking decision of when to use automation testing and manual testing based on the context to be tested.

Course Outcomes:

- Ability to design and develop the best test strategies in accordance to the development models.
- Acquire skills to perform dataflow testing, domain testing, logic testing.
- Acquire Exposure to develop KV charts, anomaly detection.
- Acquire knowledge on various types of state graphs.
- Ability to use suitable software testing tools and discuss their range and capabilities.
- Ability to apply performance testing for database applications.

UNIT I

Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs. STLC (software testing life cycle) phases.

Flow graphs and Path testing: - Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT II

Transaction Flow Testing:-transaction flows, transaction flow testing techniques.

Dataflow testing: - Basics of data flow testing, strategies in data flow testing, application of dataflow testing.

Domain Testing:-domains and paths, nice & ugly domains, domain testing, domain and interface testing, domains and testability.

UNIT III

Paths, Path products and Regular expressions: - path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

Logic Based Testing: - overview, decision tables, path expressions, kv charts, specifications.

UNIT IV

State, State Graphs and Transition testing: - state graphs, good & bad state graphs, state testing, Testability tips, Automation Testing versus Manual Flow Testing.

UNIT V

Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, creation of test script for unattended testing, synchronization of test case, Rapid testing, Performance testing of a data base application and HTTP connection for website access, node reduction algorithm, building tools. Student should be given an exposure to Selenium, JMeter, QTP, Sahi automation testing tools.

Text Books:

1. Software Testing techniques - BarisBeizer, Dreamtech, second edition.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

References:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD (Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

21CS3274: DISTRIBUTED DATABASES
(PROFESSIONAL ELECTIVE-II)

III B. Tech II Sem. **L T P C**
3 - - 3

Prerequisites

- A course on "Database Management Systems"

Course Objectives:

- The purpose of the course is to enrich the previous knowledge of database systems and
- Exposing the need for distributed database technology to confront with the deficiencies of the centralized database systems.
- Introduce basic principles and implementation techniques of distributed database systems.
- Equip students with principles and knowledge of parallel and object-oriented databases.
- Topics include distributed DBMS architecture and design; query processing and optimization.
- Distributed transaction management and reliability; parallel and object database management systems.

Course Outcomes:

- Understand theoretical and practical aspects of distributed database systems.
- Study and identify various issues related to the development of distributed database system.
- Understand the design aspects of object-oriented database system and relate development.
- Able to Practice Parallel distributed databases.
- Identify the differences between OODBMS and ORDBMS.

UNIT I

Introduction: Distributed Data Processing, Distributed Database System, Promises of DDBSs, Problem areas. Distributed DBMS Architecture: Architectural Models for Distributed DBMS, DDMBS Architecture.

Distributed Database Design: Alternative Design Strategies, Distribution Design issues, Fragmentation, Allocation.

UNIT II

Query processing and decomposition: Query processing objectives, characterization of query processors, layers of query processing, query decomposition, localization of distributed data.

Distributed query Optimization: Query optimization, centralized query optimization, distributed query optimization algorithms.

UNIT III

Transaction Management: Definition, properties of transaction, types of transactions.

Distributed Concurrency Control: serializability, concurrency control mechanisms & algorithms, time - stamped & optimistic concurrency control Algorithms, deadlock Management.

UNIT IV

Distributed DBMS Reliability: Reliability concepts and measures, fault-tolerance in distributed systems, failures in Distributed DBMS, local & distributed reliability protocols, site failures and network partitioning.

Parallel Database Systems: Parallel database system architectures, parallel data placement, parallel query processing, load balancing, database clusters.

UNIT V

Distributed object Database Management Systems: Fundamental object concepts and models, object distributed design, architectural issues, object management, distributed object storage, object query Processing.

Object Oriented Data Model: Inheritance, object identity, persistent programming languages, persistence of objects, comparison OODBMS and ORDBMS.

Text Books:

1. M. Tamer OZSU and Patuck Valduriez: Principles of Distributed Database Systems, Pearson Edn. Asia, 2001.
2. Stefano Ceri and Giuseppe Pelagatti: Distributed Databases, McGraw Hill.

References:

1. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom: "Database Systems: The Complete Book", Second Edition, Pearson International Edition
2. Chanda Ray (2012), Distributed Database Systems, 1st Edition, Pearson Education India.

21IT3251: CRYPTOGRAPHY AND NETWORK SECURITY LAB

III B. Tech II Sem. **L T P C**

- - 2 1.5

Prerequisites:

- Should have knowledge on c programming and java programming

Course Objectives:

- To understand basics operations involved in encryption and decryption algorithms.
- To be able to implement Conventional Encryption Algorithms.
- To be able to implement Public Key Cryptography Algorithms.
- To be able to implement Message Authentication Algorithms.
- To be able to implement cryptographic Hash Functions algorithms.

Course Outcomes:

- Student will be able to understand basic mathematical operations involved in cryptography.
- Student will be able to implement Conventional Encryption Algorithms i.e. DES, AES, Blowfish, RC4 algorithms.
- Student will be able to implement Public Key Cryptography Algorithms i.e. RSA, Diffie-Hellman algorithms
- Student will be able to implement Message Authentication Algorithms.
- Student will be able implement cryptographic Hash Functions algorithms.

EXPERIMENTS:

1. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a Java program to perform encryption and decryption using the following algorithms a. Ceaser cipher b. Substitution cipher c. Hill Cipher
4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA.

Text Books:

1. Cryptography and Network Security- Principles and Practice: William Stallings, Pearson Education, 6th Edition
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition

References:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr. T R Padmanabhan, Wiley India, 1st Edition.
2 Cryptography and Network Security: Forouzan Mukhopadhyay, Mc Graw Hill, 3rd Edition

21IT3252: COMPILER DESIGN LAB

III B. Tech II Sem. **L T P C**
- - 3 1.5

Prerequisites:

- Should have knowledge on C programming

Course Objectives

- To understand the various phases in the design of a compiler.
- To understand syntax directed translation schemes
- To introduce Lex and YACC tools.
- To implement Top-down parsing technique.
- To implement Bottom-up parsing technique.

Course Outcomes:

- Ability to design, develop, and implement a compiler for any language.
- Ability to use different tools in construction of the phases of a compiler for the mini language.
- Ability to implement Lexical Analyzer for given language using C and Lex tool.
- Ability to use YACC tools for developing a parser.
- Able to design and implement LL and LR parsers.

List of experiments:

1. Design a lexical analyser for given language and the lexical analyser should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language.

2. a. Write a C program to identify whether a given line is a comment or not

   b.Write a C program to test whether a given identifier is valid or not.

3. Write a C program to simulate lexical analyzer for validating operators

4. To Study about Lexical Analyzer Generator (LEX) and Flex (Fast Lexical Analyzer)

5. Implement following programs using Lex.

   a. Create a Lexer to take input from text file and count no of characters, no. of lines & no. of words.

   b. Write a Lex program to count number of vowels and consonants in a given input string.

6. Implement following programs using Lex.

   a. Write a Lex program to print out all numbers from the given file.

   b. Write a Lex program to printout all HTML tags in file.c.

7. Write a Lex program which adds line numbers to the given file and display the same onto the standard output.

8. Write a C program for constructing of LL (1) parsing.

9. Write a C program for constructing recursive descent parsing

10. Write a C program to implement LALR parsing.

Text Books:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman.

References:

1. Lex&Yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
2. Compiler Construction, Louden, Thomson.

21CS3253: MACHINE LEARNING LAB

III B. Tech II Sem. **L T P C**

- - 3 1.5

Prerequisites:

- Knowledge of Java Programming.
- Data mining concepts.

Course Description:

- Machine Learning is concerned with computer programs that automatically improve their

performance through experience.

- This course covers the theory and practical algorithms for machine learning from a variety of perspectives.
- This course covers topics such as FIND-S, Candidate Elimination Algorithm, Decision tree (ID3 Algorithm), Back propagation Algorithm, Naïve Bayesian classifier, Bayesian Network, k-Means Algorithm, k-Nearest Neighbor Algorithm, Locally Weighted Regression Algorithm.

Course Objectives:

- Make use of Data sets in implementing the machine learning algorithms
- Implement the machine learning concepts and algorithms in any suitable language of choice.

Course Outcomes:

- Understand the implementation procedures for the machine learning algorithms
- Design Java/Python programs for various Learning algorithms.
- Apply appropriate data sets to the Machine Learning algorithms
- Identify and apply Machine Learning algorithms to solve real world problems

List of Programs:

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples
3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample
4. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Text Books:

1. Machine Learning, Tom M Michel, McGraw Hill, 1997.

References:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis
2. https://towardsdatascience.com/tagged/model - evaluation
3. https://github.com/topics/handwriting-recognition?l=python

21MC0005: INDIAN CONSTITUTION

B. Tech. III Year I Sem **L T P C**

3 0 0 0

The Constitution of India is the supreme law of India. Parliament of India cannot make any law which violates the Fundamental Rights enumerated under the Part III of the Constitution. The Parliament of India has been empowered to amend the Constitution under Article 368, however, it cannot use this power to change the "basic structure" of the constitution, which has been ruled and explained by the Supreme Court of India in its historical judgments.

The Constitution of India reflects the idea of "Constitutionalism" –a modern and progressive concept historically developed by the thinkers of "liberalism" –an ideology which has been recognized as one of the most popular political ideology and result of historical struggles against arbitrary use of sovereign power by state. The historic revolutions in France, England, America and particularly European Renaissance and Reformation movement have resulted into progressive legal reforms in the form of "constitutionalism" in many countries. The Constitution of India was made by borrowing models and principles from many countries including United Kingdom and America.

The Constitution of India is not only a legal document but it also reflects social, political and economic perspectives of the Indian Society. It reflects India's legacy of "diversity". It has been said that Indian constitution reflects ideals of its freedom movement; however, few critics have argued that it does not truly incorporate our own ancient legal heritage and cultural values. No lawcan be "static" and therefore the Constitution of India has also been amended more than one hundred times. These amendments reflect political, social and economic developments since the year 1950.

The Indian judiciary and particularly the Supreme Court of India has played an historic role as the guardian of people. It has been protecting not only basic ideals of the Constitution but also strengthened the same through progressive interpretations of the text of the Constitution. The judicial activism of the Supreme Court of India and its historic contributions has been recognizedthroughout the world and it gradually made it "as one of the strongest court in the world".

COURSE CONTENT:

1. Meaning of the constitution law and constitutionalism
2. Historical perspective of the Constitution of India
3. Salient features and characteristics of the Constitution of India
4. Scheme of the fundamental rights
5. The scheme of the Fundamental Duties and its legal status

6. The Directive Principles of State Policy –Its importance and implementation
7. Federal structure and distribution of legislative and financial powers between the Unionand the States
8. Parliamentary Form of Government in India –The constitution powers and status of thePresident of India
9. Amendment of the Constitutional Powers and Procedure
10. The historical perspectives of the constitutional amendments in India
11. Emergency Provisions: National Emergency, President Rule, Financial Emergency
12. Local Self Government –Constitutional Scheme in India
13. Scheme of the Fundamental Right to Equality
14. Scheme of the Fundamental Right to certain Freedom under Article 19
15. Scope of the Right to Life and Personal Liberty under Article 21

21MC0007: YOGA & INDIAN PHILOSOPHY

III B. Tech II Sem. **L T P C**
3 - - 0

UNIT I
Bhagavad Gita, chapter 2 Sankhya Yoga slokas 54-72 about emotional intelligence (Stitaprajnata)

UNIT II
Bhagavad Gita, chapters 3-7

UNIT III
Bhagavad Gita, chapters 8-11

UNIT IV
Bhagavad Gita, chapters 12-15

UNIT V
Bhagavad Gita, chapters 16-18
10 quotes from each chapter of ref.(2)

References:
1) Bhagavad Gita By Swami Swarupananda, R K Math Publication
2) Vivekananda-His Call to the Nation, R K Math Publication