

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**  
**B.Tech (R21) COURSE STRUCTURE**  
**III YEAR I-SEMESTER**

S. No.	Course Code	Course Title	Category	L	T	P	C
1	21CS3111	Computer Networks	PC	3	-	-	3
2	21CS3112	Software Engineering	PC	3	-	-	3
3	21CS3113	Web Technologies	PC	3	-	-	3
<b>Professional Elective – I</b>							
4	21CS3171	Data Analytics	PE	3	-	-	3
	21CS3172	Advanced Computer Architecture	PE				
	21CS3173	Digital Image Processing	PE				
	21CS3174	Distributed Systems	PE				
5	<b>Open Elective – I</b>		OE	3	-	-	3
<b>PRACTICAL</b>							
6	21CS3151	Computer Networks & Unified Modeling Language 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	21CS3181	Summer Internship	PW		-	-	1
10	21MC0006	Aptitude and Logical Reasoning	MC	3	-	-	0
<b>Total</b>				<b>18</b>	<b>-</b>	<b>8</b>	<b>20</b>

**III YEAR II- SEMESTER**

S.No.	Course Code	Course Title	Category	L	T	P	C
1	21CS3211	Cryptography and Network Security	PC	3	-	-	3
2	21IT3211	Compiler Design	PC	3	-	-	3
3	21CS3213	Machine Learning	PC	3	-	-	3
<b>Professional Elective – II</b>							
4	21CS3271	Artificial Intelligence	PE	3	-	-	3
	21CS3272	Software Project Management	PE				
	21CS3273	Information Retrieval Systems	PE				
	21CS3274	Distributed Databases	PE				
5	<b>Open Elective – II</b>		OE	3	-	-	3
<b>PRACTICAL</b>							
6	21CS3251	Cryptography and Network Security Lab	PC	-	-	3	1.5
7	21CS3252	Compiler Design Lab	PC	-	-	3	1.5
8	21CS3253	Machine Learning Lab	PC	-	-	3	1.5
9	21MC0007	Yoga and Indian Philosophy	MC	3	-	-	0
<b>Total</b>				<b>18</b>	<b>-</b>	<b>9</b>	<b>19.5</b>

# 21CS3111: COMPUTER NETWORKS

**B.Tech. III Year I Sem.**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>3</b>	<b>-</b>	<b>-</b>	<b>3</b>

## **Prerequisites**

- A course on “Programming for problem-solving”
- A course on “Data Structures”

## **Course Objectives**

- To introduce an overview of the concepts and fundamentals of computer networks.
- To Study Data Link Layer Concepts, Design issues, and protocols.
- Familiarize the working mechanism of the network layer.
- Understanding of Transport Layer Concepts and Protocols.
- To Explore the concepts of DNS,E-Mail, WWW, and various application layer Protocols.

## **Course Outcomes**

- Gain knowledge of basic computer network technology.
- Gain knowledge of the functions of each layer in the OSI and TCP/IP reference model.
- Obtain skills in subnetting and routing mechanisms.
- Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation.
- Gain Knowledge of E-Mail Protocols, Streaming Audio, and Video on the Web.

## **UNIT – I:**

**Introduction:** Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.

**Physical Layer:** Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless transmission.

## **UNIT – II :**

**Data link layer:** Design Issues, Framing, Error Detection, and Correction.

**Elementary data link protocols:** simplex protocol, A simplex stop and wait-for protocol for an error-free channel.

**Sliding Window protocols:** A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat.

**Medium Access sub layer:** The Channel Allocation Problem, Multiple Access Protocols: ALOHA, Carrier Sense Multiple Access Protocols.

## **UNIT – III :**

**Network Layer:** Design issues, Routing Algorithms: Shortest Path Routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service.

## **UNIT – IV:**

**Transport Layer:** Transport Services, Elements of Transport protocols, Congestion Control, Connection management, TCP and UDP protocols.

**UNIT – V:**

**Application Layer** –Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.

**TEXT BOOK:**

1. Computer Networks - Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI.

**REFERENCES:**

1. An Engineering Approach to Computer Networks-S. Keshav, 2 nd Edition, Pearson Education.
2. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH.

# 21CS3112: SOFTWARE ENGINEERING

**B.Tech. III Year 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 GrawHill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rumbaugh, 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-HillCompanies.

# 21CS3113: WEB TECHNOLOGIES

B.Tech. III Year I Sem.

L T 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.
- Analyse what is XML and how to parse and use XML data with JAVA
- Develop Server side Application 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:** Introduction to 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 like 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 2<sup>nd</sup> edition, Wiley Dreamtech
2. Java Server Pages —Hans Bergsten, SPD O'Reilly,
3. Java Script, D.Flanagan
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.



# 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 Millway 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

## 21CS3172: ADVANCED COMPUTER ARCHITECTURE

**B.Tech. III Year I Sem.**

**L T P C**

**3 0 0 3**

**Prerequisites:** Computer Organization.

### **Course Objectives:**

- To impart the concepts and principles of parallel and advanced computer architectures.
- To develop the design techniques of Scalable and multithreaded Architectures.
- To Apply the concepts and techniques of parallel and advanced computer architectures to design modern computer systems.
- It will introduce students to advanced aspects of processor design and will specifically focus on out-of-order pipelines, GPUs, and compiler techniques for enhancing ILP.
- The course will subsequently move on to cache design and main memory technologies such as DDR-4.

### **Course Outcomes:** Gain knowledge of

- Computational models and Computer Architectures.
- Concepts of parallel computer models.
- Scalable Architectures, Pipelining, Superscalar processors, multiprocessors.
- Interpret the performance of a processor based on metrics such as execution time, cycles per instruction (CPI), Instruction count etc.
- Predict the challenges of realizing different kinds of parallelism (such as instruction, data, thread, core level) and leverage them for performance advancement.

### **UNIT – I**

Theory of Parallelism, Parallel computer models, The State of Computing, Multiprocessors and Multicomputers, Multivector and SIMD Computers, PRAM and VLSI models, Architectural development tracks, Program and network properties, Conditions of parallelism, Program partitioning and Scheduling, Program flow Mechanisms, System interconnect Architectures.

### **UNIT - II**

Principals of Scalable performance, Performance metrics and measures, Parallel Processing applications, Speed up performance laws, Scalability Analysis and Approaches, Hardware Technologies, Processes and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology.

### **UNIT - III**

Bus Cache and Shared memory, Backplane bus systems, Cache Memory organizations, Shared-Memory Organizations, Sequential and weak consistency models, Pipelining and superscalar techniques, Linear Pipeline Processors, Non-Linear Pipeline Processors, Instruction Pipeline design, Arithmetic pipeline design, superscalar pipeline design.

### **UNIT - IV**

Parallel and Scalable Architectures, Multiprocessors and Multi computers, Multiprocessor system interconnects, cache coherence and synchronization mechanism, Three Generations of Multi computers, Message-passing Mechanisms, Multi vector and SIMD computers, Vector

Processing Principals, Multi vector Multiprocessors, Compound Vector processing, SIMD computer Organizations, The connection machine CM-5,

#### **UNIT - V**

Scalable, Multithreaded and Dataflow Architectures, Latency-hiding techniques, Principals of Multithreading, Fine-Grain Multi computers, Scalable and multithreaded Architectures, Dataflow and hybrid Architectures.

#### **Text Book**

1. Advanced Computer Architecture, Kai Hwang, 2<sup>nd</sup> Edition, Tata McGraw Hill Publishers.

#### **References:**

1. Computer Architecture, J.L. Hennessy and D.A. Patterson, 4<sup>th</sup> Edition, ELSEVIER.
2. Advanced Computer Architectures, S.G.Shiva, Special Indian edition, CRC, Taylor & Francis.
3. Introduction to High Performance Computing for Scientists and Engineers, G. Hager and G. Wellein, CRC Press, Taylor & Francis Group.
4. Advanced Computer Architecture, D. Sima, T. Fountain, P. Kacsuk, Pearson education.
5. Computer Architecture, B. Parhami, Oxford Univ. Press.

# 21CS3173: DIGITAL IMAGE PROCESSING

(Professional Elective - I)

B.Tech. III Year I Sem.

L T P C

3 - - 3

## Prerequisites

- Students are expected to have knowledge in linear signals and systems, Fourier Transform, basic linear algebra, basic probability theory and basic programming techniques; knowledge of Digital Signal Processing is desirable.
- A course on “Computational Mathematics”
- A course on “Computer Oriented Statistical Methods”

## Course Objectives

- Provide a theoretical and mathematical foundation of fundamental Digital Image Processing concepts.
- The topics include image acquisition; sampling and quantization; preprocessing; enhancement; restoration; segmentation; and compression.
- To introduce the concepts of image processing and basic analytical methods to be used in image processing.
- To familiarize students with image enhancement and restoration techniques.
- To explain different Image compression techniques.

## Course Outcomes

- Demonstrate the knowledge of the basic concepts of two-dimensional signal acquisition, sampling, and quantization.
- Demonstrate the knowledge of filtering techniques.
- Demonstrate the knowledge of 2D transformation techniques.
- Demonstrate the knowledge of image enhancement, segmentation, restoration and compression techniques.
- Demonstrate the knowledge of Image Compression Models.

## UNIT-I:

Fundamentals of Image processing and Image Transforms: Basic steps of Image processing system sampling and quantization of an Image – Basic relationship between pixels Image Transforms: 2 – D Discrete Fourier Transform, Discrete Cosine Transform (DCT), Discrete Wavelet transforms

## UNIT-II:

Image Processing Techniques: Image Enhancement: Spatial Domain methods: Histogram Processing, Fundamentals of Spatial Filtering, Smoothing Spatial filters, Sharpening Spatial filters Frequency Domain methods: Basics of filtering in frequency domain, image smoothing, image sharpening, selective filtering Image Segmentation: Segmentation concepts, point, line and Edge detection, Thresholding, region based segmentation

## UNIT-III:

Image Compression Image compression fundamentals – coding Redundancy, spatial and temporal redundancy. Compression models : Lossy and Lossless, Huffmann coding, Arithmetic

coding, LZW coding, run length coding, Bit Plane coding, transform coding, predictive coding , wavelet coding, JPEG standards.

#### **UNIT-IV:**

Basic Steps of Video Processing: Analog video, Digital Video, Time varying Image Formation models: 3D motion models, Geometric Image formation , Photometric Image formation, sampling of video signals, filtering operations .

#### **UNIT-V:**

2-D Motion Estimation: Optical flow, general methodologies, pixel based motion estimation, Block matching algorithm, Mesh based motion Estimation, global Motion Estimation, Region based motion estimation, multi resolution motion estimation. Waveform based coding, Block based transform coding, predictive coding, Application of motion estimation in video coding.

#### **TEXT BOOKS**

1. Gonzalez and Woods ,”Digital Image Processing “, 3rd edition , Pearson
2. Yao wang, Joem Ostarmann and Ya – quin Zhang “Video processing and communication 1st edition , PHI

#### **REFERENCE TEXT BOOKS**

1. M. Tekalp ,”Digital video Processing”, Prentice Hall International
2. Relf, Christopher G.,”Image acquisition and processing with LabVIEW”, CRC press
3. Aner ozdemi R, "Inverse Synthetic Aperture Radar Imaging with MATLAB Algorithms", John Wiley & Sons
4. Chris Solomon, Toby Breckon ,”Fundamentals of Digital Image Processing A Practical Approach with Examples in Matlab”, John Wiley & Sons,

# **21CS3174: DISTRIBUTED SYSTEMS**

**(Professional Elective - I)**

**B.Tech. III Year I Sem.**

**L T P C**

**3 - - 3**

## **COURSE OBJECTIVES**

- To understand the foundations of distributed systems.
- To learn issues related to clock Synchronization and the need for global state in distributed systems.
- To understand the significance of agreement, fault tolerance and recovery protocols in Distributed Systems.
- To introduce the concepts of distributed file systems, shared memory and message passing systems, synchronization and resource management.
- To introduce the concepts of Transactions and Concurrency Control and Distributed deadlocks

## **COURSE OUTCOMES:**

- Distinguish distributed computing paradigm from other computing paradigms.
- Able to explain various distributed algorithms, such as logical clocks and leader election.
- Illustrate the mechanisms of inter process communication in distributed system.
- Explain name services and distributed shared memory.
- The students will be able to define, explain and illustrate fundamental principles of concurrent transaction processing.

### **UNIT-I**

Characterization of Distributed Systems: Introduction, Examples of Distributed systems, Resource sharing and web, challenges.

System Models: Introduction, Architectural and Fundamental models.

### **UNIT-II**

Time and Global States: Introduction, Clocks, Events and Process states, synchronizing physical clocks, Logical time and Logical clocks, Global states, Distributed Debugging.

Coordination and Agreement: Introduction, Distributed mutual exclusion, Elections, Multicast Communication, Consensus and Related problems.

### **UNIT-III**

Inter Process Communication: Introduction, The API for the internet protocols, External Data Representation and Marshalling, Client-Server Communication, Group Communication, Case Study: IPC in UNIX.

Distributed Objects and Remote Invocation: Introduction, Communication between Distributed Objects, Remote Procedure Call, Events and Notifications, Case study-Java RMI.

#### **UNIT-IV**

Distributed File Systems: Introduction, File service Architecture, Case Study1: Sun Network File System, Case Study 2: The Andrew File System.

Name Services: Introduction, Name Services and the Domain Name System, Directory Services, Case study of the Global Name Service.

Distributed Shared Memory: Introduction Design and Implementation issues, Sequential consistency and Ivy case study, Release consistency and Munin case study, other consistency models.

#### **UNIT-V**

Transactions and Concurrency Control: Introduction, Transactions, Nested Transactions, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control.

Distributed Transactions: Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery

#### **TEXT BOOK:**

1. Distributed Systems, Concepts and Design, George Coulouris, J Dollimore and TIm Kindberg, Pearson Education, 4h Edition, 2009.

#### **REFERENCE BOOKS:**

1. Distributed Systems, Principles and paradigms, Andrew S.Tanenbaum, Maarten Van
2. Steen, Second Edition, PHI.
3. Distributed Systems, An Algorithm Approach, Sikumar Ghosh, Chapman & Hall/CRC,
4. Taylor & Fransis Group, 2007.
5. Distributed Systems, Principles and Paradigms, Andrew S.Tanenbaum, Maarten Van Steen, 2d Edition, PHI.
6. Distributed Systems, An Algorithm Approach, Sukumar Ghosh,Chapman&HalyCRC, Taylor & Fransis Group, 2007.



# 21CS3151: COMPUTER NETWORKS & UNIFIED MODELING LANGUAGE LAB

B.Tech. III Year I Sem.

L T P C  
- - 3 1.5

## Course Objectives

- To understand the various phases in the design of a compiler & to understand the design of top-down and bottom-up parsers.
- To understand the working principle of various communication protocols.
- To analyze the traffic flow and the contents of protocol frames
- To have hands on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.
- Learn Syntax and Semantics and create Functions in Python, Handle Strings and Files in Python.

## Course Outcomes

- Implement data link layer framing methods , analyze error detection and error correction codes
- Implement and analyze routing and congestion issues in network design
- Implement Encoding and Decoding techniques used in presentation layer
- The students should be able to examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
- Demonstrate proficiency in handling Strings and File Systems.

## COMPUTER NETWORKS

### List of Experiments:

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijkstra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Write a program for congestion control using Leaky bucket algorithm.
8. Write a program for frame sorting technique used in buffers.
9. Programs for IP address conversion function
10. Write a client-server application for chat using UDP
  - i. Wire shark
  - ii. Packet Capture Using Wire shark
  - iii. Starting Wire shark
  - iv. Viewing Captured Traffic
  - v. Analysis and Statistics & Filters

## UNIFIED MODELLING

### Tools:

- Use smart draw for data flow diagrams

- Rational Rose or Star UML for UML diagrams

### **LIST OF EXPERIMENTS:**

Do the following exercises for projects given in the list of sample projects

1. Development of problem statement.
2. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3. Draw level 0, level 1, and level 2 dataflow diagrams
4. Study and usage of any Design phase CASE tool
5. Performing the Design by using any Design phase CASE tools.
6. Performing functional testing and create simple testing report

### **Sample Projects:**

1. Hospital management system
2. Online mobile recharge portal
3. Online Exam Registration
4. Stock Maintenance System
5. E-ticketing for Travel system
6. Credit Card Processing
7. E-book management System.
8. Online Recruitment system

### **TEXT BOOK:**

1. Computer Networks - Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI.

### **REFERENCES:**

1. An Engineering Approach to Computer Networks-S. Keshav, 2 nd Edition, Pearson Education.
2. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH.

## 21CS3152: WEB TECHNOLOGIES LAB

B.Tech. III Year I Sem.

L T P C  
- - 3 1.5

### Course Objectives

- To enable the student to program web applications using the following technologies HTML, Java script , 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

### 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](http://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](mailto: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, 8th Edition
5. Web technologies, Black Book, Dreamtech press.

# **21HS3153: ADVANCED ENGLISH COMMUNICATION SKILLS LAB**

**B.Tech. III Year 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

B.Tech. III Year 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. Quantitative 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 given 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 Competitive Examinations, U.Mohan Rao, Scitech Publication.

# 21CS3211: CRYPTOGRAPHY & NETWORK SECURITY

B.Tech. III Year II Sem.

L	T	P	C
3	-	-	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

**B.Tech. III Year 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, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, 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 Chandrashekar, 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

B.Tech. III Year 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 BOOK:**

1. Machine Learning – Tom M. Mitchell, - MGH.
2. Introduction to Machine Learning with Python, Author – Andreas C. Müller, Sara h Guido, Edition – First Edition, Publisher – O'Reilly Media, Inc.

#### **REFERENCE BOOK:**

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis.
2. Mathematics for Machine learning, Author – Marc Peter Deisenroth, Edition – First Edition, Publisher – Cambridge University Press.

# 21CS3271: ARTIFICIAL INTELLIGENCE

B.Tech. III Year II Sem.

L T P C

3 - - 3

## Course Objectives

- To train the students to understand different types of AI agents.
- Various AI search algorithms.
- Fundamentals of knowledge representation.
- To apply knowledge representation, reasoning.
- Study of Markov Models enable the student ready to step into applied AI.

## Course Outcomes

- Understand AI problems and problem solving agents and search strategies
- Apply advanced search techniques and acquire basic knowledge representation and reasoning logic.
- Apply reasoning under uncertainty.
- Understand learning strategies.
- Understand implementation of expert system.

## UNIT - I

**Introduction:** AI problems, Agents and Environments, Structure of Agents, Problem Solving Agents Basic Search Strategies: Problem Spaces, Uninformed Search (Breadth-First, Depth-First Search, Depth-first with Iterative Deepening), Heuristic Search (Hill Climbing, Generic Best-First, A\*), Constraint Satisfaction (Backtracking, Local Search).

## UNIT - II

**Advanced Search:** Constructing Search Trees, Stochastic Search, A\* Search Implementation, Minimax Search, Alpha-Beta Pruning.

**Basic Knowledge Representation and Reasoning:** Propositional Logic, First-Order Logic, Forward Chaining and Backward Chaining, Introduction to Probabilistic Reasoning, Bayes Theorem

## UNIT - III

**Advanced Knowledge Representation and Reasoning:** Knowledge Representation Issues, Non monotonic Reasoning, Other Knowledge Representation Schemes

**Reasoning Under Uncertainty:** Basic probability, Acting Under Uncertainty, Bayes' Rule, Representing Knowledge in an Uncertain Domain, Bayesian Networks.

## UNIT - IV

**Learning:** What Is Learning? Rote Learning, Learning by Taking Advice, Learning in Problem Solving, Learning from Examples, Winston's Learning Program, Decision Trees.

## UNIT - V

**Expert Systems:** Representing and Using Domain Knowledge, Shell, Explanation, Knowledge Acquisition.

## TEXT BOOK:

1. Russell, S. and Norvig, P, Artificial Intelligence: A Modern Approach, Third Edition, PrenticeHall, 2010.

## REFERENCES:

1. Artificial Intelligence, Elaine Rich, Kevin Knight, Shivasankar B. Nair, The McGraw Hill publications, Third Edition, 2009.
2. George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education, 6th ed., 2009.

# 21CS3272: SOFTWARE PROJECT MANAGEMENT

(Professional Elective - II)

B.Tech. III Year II Sem.

L T P C

3 - - 3

## Course Objectives

- A basic knowledge of software project management principles
- The ability to come up with a project schedule and assign resources
- Choose an appropriate project development methodology (e.g. waterfall, spiral ...)
- Identify project risks, monitor and track project deadlines.
- The capability to work in a team environment and be aware of different modes of communications.

## Course Outcomes

- Identify and describe how different project contexts will impact upon all aspects of a software development project
- Identify and describe the key phases of project management and the key skills associated with each.
- Determine an appropriate project management approach through an evaluation of the business context and project scope and knowledge of agile and traditional project management approaches.
- Demonstrate through application, knowledge of the key project management skills, such as product and work break-down structure, schedule; governance including progress reporting, risk and quality management.
- As part of a small team research and produce a concise piece of writing suitable for presentation to senior management.

## UNIT-I

**Conventional Software Management:** The waterfall model, conventional software Management performance. Overview of Project Planning – Stepwise Project Planning.

**Improving Software Economics:** Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

## UNIT-II

**The old way and the new way:** The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

**Life cycle phases:** Engineering and production stages, Inception, Elaboration, Construction, Transition phases.

**Artifacts of the process:** The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

## UNIT –III

**Work Flows of the process:** Software process workflows, Iteration workflows. **Checkpoints of the process:** Major milestones, Minor Milestones, Periodic status assessments.

**Iterative Process Planning:** Work breakdown structures, planning guidelines, cost and



schedule estimating, Iteration planning process, Pragmatic planning.

#### **UNIT-IV**

**Process Automation:** Automation Building blocks.

**Project Control and Process instrumentation:** These Vendor Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

**Tailoring the Process:** Process discriminants.

#### **UNIT-V**

**Project Organizations and Responsibilities:**

Line-of-Business Organizations, Understanding Behavior – Organizational Behavior

**Future Software Project Management:** Modern Project Profiles, Next generation Software economics, modern process transitions.

**Case Study:** The command Center Processing and Display system-Replacement (CCPDS-R).

#### **TEXT BOOKS:**

1. Software Project Management, Walker Royce: Pearson Education,2005.

#### **REFERENCES:**

1. Software Project Management, Bob Hughes and Mike Cotterell:Tata McGraw-HillEdition.
2. Software Project Management, Joel Henry, Pearson Education.  
Software Project Management in practice, Pankaj Jalote, Pearson Education.2

# 21CS3273: INFORMATION RETRIEVAL SYSTEMS

(Professional Elective-II)

B.Tech. III Year II Sem.

L T P C

3 - - 3

## Course Objectives

- To learn the important concepts and algorithms in IRS
- To understand the data/file structures that is necessary to design, and implement information retrieval (IR) systems.
- Demonstrate Information visualization technologies like Cognition and perception in the Internet or Web search engine.
- Analyze ranked retrieval of a very large number of documents with hyperlinks between them.
- Describe hands-on experience store, and retrieve information from www using semantic approaches.

## Course Outcomes

- Ability to understand IR principles to locate relevant information in large collections of data
- Ability to understand information extraction using indexing, and various data structure algorithms.
- Ability to design different document clustering algorithms and understand automatic indexing.
- Ability to use various search algorithms and perform information visualization.
- Ability to understand the ways to design an Information Retrieval System for web search tasks.

## UNIT – I

**Introduction to Information Retrieval Systems:** Definition of Information Retrieval System, Objectives of Information Retrieval Systems, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses.

**Information Retrieval System Capabilities:** Search Capabilities, Browse Capabilities, Miscellaneous Capabilities.

## UNIT - II

**Cataloging and Indexing:** History and Objectives of Indexing, Indexing Process, Automatic Indexing, Information Extraction.

**Data Structure:** Introduction to Data Structure, Stemming Algorithms, Inverted File Structure, N-Gram Data Structures, PAT Data Structure, Signature File Structure, Hypertext and XML Data Structures, Hidden Markov Models.

## UNIT - III

**Automatic Indexing:** Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages.

**Document and Term Clustering:** Introduction to Clustering, Thesaurus Generation, Item Clustering, Hierarchy of Clusters.

## **UNIT - IV**

**User Search Techniques:** Search Statements and Binding, Similarity Measures and Ranking, Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext.

**Information Visualization:** Introduction to Information Visualization, Cognition and Perception, Information Visualization Technologies.

## **UNIT - V**

**Text Search Algorithms:** Introduction to Text Search Techniques, Software Text Search Algorithms, Hardware Text Search Systems.

**Multimedia Information Retrieval:** Spoken Language Audio Retrieval, Non-Speech Audio Retrieval, Graph Retrieval, Imagery Retrieval, Video Retrieval, Web search basics. Web crawling and indexes.

## **TEXT BOOKS:**

1. Information Storage and Retrieval Systems – Theory and Implementation, Second Edition, Gerald J. Kowalski, Mark T. Maybury, Springer.

## **REFERENCES:**

1. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms, Prentice Hall, 1992.
2. Information Storage & Retrieval By Robert Korfhage – John Wiley & Sons.
3. Modern Information Retrieval By Yates and Neto Pearson Education.

# 21CS3274: DISTRIBUTED DATABASES

(Professional Elective-II)

B.Tech. III Year 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 related 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.

# 21CS3251: CRYPTOGRAPHY AND NETWORK SECURITY LAB

**B.Tech. III Year II Sem.**

**L T P C**

**- - 3 1.5**

## **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 to 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
4. a. Ceaser cipher      b. Substitution cipher    c. playfair cipher
5. Write a C/JAVA program to implement the DES 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.

## 21CS3252: COMPILER DESIGN LAB

**B.Tech. III Year II Sem.**

**L T P C**

**- - 3 1.5**

### **Course Objectives**

- To understand the various phases in the design of a compiler.
- To introduce LEX tool.
- To Understand Top down and Bottom up Parsers.
- To design and implement basic techniques of compiler construction and tools that can used to perform syntax-directed translation of a high-level programming language into an executable code.
- To provide deeper insights into the more advanced semantics aspects of programming languages, code generation, machine independent optimizations, dynamic memory allocation, and object orientation.

### **Course Outcomes**

- To use LEX compiler for developing a scanner.
- To Design and implement a lexical analyzer.
- To Implement the Intermediate code generation and Storage allocation strategy.
- To Design and implement LL and LR Parsers.
- To design, develop, and implement a compiler for any language.

### **List of experiments:**

1. Design a lexical analyzer for given language and the lexical analyzer 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 BOOK:**

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

**REFERENCES:**

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

# 21CS3253: MACHINE LEARNING LAB

**B.Tech. III Year II Sem.**

**L T P C**

**- - 3 1.5**

## **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 BOOK:**

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

**REFERENCE BOOKS:**

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

## **21MC0006: YOGA AND INDIAN PHILOSOPHY**

**B.Tech. III Year II Sem.**

**L T P C**

**2 - - 0**

### **Unit-1**

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

### **Unit-2**

Bhagavad Gita, chapters 3-7

### **Unit-3**

Bhagavad Gita, chapters 8-11

### **Unit-4**

Bhagavad Gita, chapters 12-15

### **Unit-5**

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