Faculty of Engineering

Syllabus

T.E. (Information Technology) 2012 Course

(With effect from Academic Year 2014 - 15)

UNIVERSITY OF PUNE

THE SYLLABUS IS PREPARED BY:

B.O.S. in Information Technology, University of Pune

PROGRAM EDUCATIONAL OBJECTIVES

The students of Information Technology course after passing out will

1. Graduates of the program will possess strong fundamental concepts in mathematics, science, engineering and Technology to address technological challenges.

2. Possess knowledge and skills in the field of Computer Science & Engineering and Information Technology for analyzing, designing and implementing complex engineering problems of any domain with innovative approaches.

3. Possess an attitude and aptitude for research, entrepreneurship and higher studies in the field of Computer Science & Engineering and Information Technology.

4. Have commitment to ethical practices, societal contributions through communities and life-long learning.

5. Possess better communication, presentation, time management and team work skills leading to responsible & competent professionals and will be able to address challenges in the field of IT at global level.

PROGRAM OUTCOMES

The students in the Information Technology course will attain:

1. an ability to apply knowledge of computing, mathematics including discrete mathematics as well as probability and statistics, science, and engineering and technology;

2. an ability to define a problem and provide a systematic solution with the help of conducting experiments, as well as analyzing and interpreting the data;

3. an ability to design, implement, and evaluate a software or a software/hardware system, component, or process to meet desired needs within realistic constraints;

4. an ability to identify, formulate, and provide systematic solutions to complex engineering problems;

5. an ability to use the techniques, skills, and modern engineering technologies tools, standard processes necessary for practice as a IT professional;

6. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems with necessary constraints and assumptions;

7. an ability to analyze the local and global impact of computing on individuals, organizations and society;

8. an ability to understand professional, ethical, legal, security and social issues and responsibilities;

9. an ability to function effectively as an individual or as a team member to accomplish a desired goal(s);

10. an ability to engage in life-long learning and continuing professional development to cope up with fast changes in the technologies/tools with the help of electives, professional organizations and extra-curricular activities;

11. an ability to communicate effectively in engineering community at large by means of effective presentations, report writing, paper publications, demonstrations;

12. an ability to understand engineering, management, financial aspects, performance, optimizations and time complexity necessary for professional practice;

13. an ability to apply design and development principles in the construction of software systems of varying complexity.

T.E.(Information Technology) 2012 Course to be implemented from June 2014

SEMESTER – I

Subject Code	Subject	Teaching Scheme			Examination Scheme					Total Marks
		Lecture	Practical	Tutorial	In-Semester Assessment	TW	PR	OR	End Semester Examination	
					Phase - I				Phase - II	
314441	Computer Network Technology	3			30				70	**100**
314442	Theory of Computation	4			30				70	**100**
314443	Database Management Systems	4			30				70	**100**
314444	Software Engineering	3			30				70	**100**
314445	Web Engineering and Technology	3			30				70	**100**
314446	Software Laboratory - I		4			50	50			**100**
314447	Database Management Systems Laboratory		4				50	50		**100**
314448	Employability Skill Development Laboratory	1	2			50				**50**
Total		**18**	**10**		**150**	**100**	**100**	**50**	**350**	**750**

Software Lab.-I   Part – I   : Assignments on Computer Network Technology and
                      Part – II   : Assignments on Web Engineering and Technology

SEMESTER – II

Subject Code	Subject	Teaching Scheme			Examination Scheme					Total Marks
		Lecture	Practical	Tutorial	In-Semester Assessment	TW	PR	OR	End Semester Examination	
					Phase - I				Phase - II	
314449	Design and Analysis of Algorithms	4			30				70	**100**
314450	Systems Programming	4			30				70	**100**
314451	Operating System	4			30				70	**100**
314452	Multimedia Technologies	3			30				70	**100**
314453	Information Tech Project Management	3			30				70	**100**
314454	Operating System Laboratory		4			50	50			**100**
314455	Software Laboratory - II		4				50	50		**100**
314456	Seminar & Technical Communication Lab.		2			50				**50**
Total		**18**	**10**		**150**	**100**	**100**	**50**	**350**	**750**

Software Lab.-II   Part – I   : Assignments on Design and Analysis of Algorithms and
                      Part – II   : Assignments on Systems Programming

SEMESTER - I

314441 : COMPUTER NETWORK TECHNOLOGY

Teaching Scheme: **Examination Scheme:**
Lectures: 3 Hours/Week   In-Semester Assessment   End-Semester Assessment
                                Phase I – 30 Marks   Phase II – 70 Marks

Prerequisites : Data Communication

Course Objectives :
1. To understand services offered by important layers of OSI model
2. To provide routing and network management techniques
3. To understand various application layer protocols and its applications in client / server environment
4. To understand various WAN technologies in computer networks

Course Outcomes :
1. Students will be able to understand the OSI model and its layer responsibilities in detail
2. Students will be able to explain various routing protocols and techniques and its related management issues at large
3. Students will be able to understand working principle of client/server application with respect to application layer protocols
4. Students will obtain thorough knowledge of various Wireless technologies

UNIT - I   NETWORK LAYER   **6 Hours**
Packet Switching, Virtual Circuits, Datagram
Routing Algorithms: Optimality Principle, Shortest path routing- Dijkstra'sAlgorithms, Distance Vector Routing, Link State Routing, Counting to infinity problem, RIP, OSPF, BGP
IP Addressing IPv4, IP Address Classes, Subnetting, CIDR/Supernetting, IP Fragmentation, ARP, DHCP, RARP, ICMP, IPv6.

UNIT - II   TRANSPORT LAYER   **6 Hours**
Transport layer duties and functionalities, application expectations and IP delivery semantics, Reliability at transport layer Vs. reliability at the Link Layer?
UDP : UDP functionality, UDP Header;
TCP : TCP Features, byte-stream, Connection-oriented, TCP Header Format, 2-way, 3-way Handshake, TCP State Diagram, TCP Sliding Window, Congestion Control Algorithms, Leaky Bucket, Token Bucket, Congestion Avoidance, RTT estimation , TCP Tahoe, Fast Retransmit, Fast Recovery,
UNIX Sockets
Congestion Avoidance, RTT Estimation , TCP Tahoe, Fast Retransmit, Fast Recovery, UNIX Sockets
Timer Management, Performance issue (concepts like: Throughput, Delay, Bandwidth Utilization, Error Rate, Congestion and Network Reliability).

UNIT - III   APPLICATION LAYER   **6 Hours**
Client/Server Model, Telnet
Domain Name System,
File Transfer protocol: FTP, TFTP,
HyperText Transfer Protocol
POP3, IMAP, SMTP, E-mail, MIME
Simple Network Management Protocol

UNIT - IV   WIRELESS LANS, PANS AND MANS   6 Hours

Introduction (Infrastructure and Ad-hoc Networks), Comparison of Wireless Networks in ISM Band
Fundamentals of WLAN – technical issues, Network Architecture, IEEE 802.11- physical layer, Mac Layer
Mechanism, CSMA/CA,   Bluetooth - Specification, Transport Layer, Middleware Protocol Group,
Bluetooth Profiles, IEEE 802.16 –differences between IEEE 802.11 and 802.16, Physical Layer, Data Link
Layer.

UNIT - V   AD-HOC NETWORKS AND SENSOR NETWORKS   6 Hours

Introduction to MANETs, Sensor Networks, Operating Environment Constraints, Protocols supported by
Wireless Networks
Applications of Sensor Networks
Sensor Node Architecture (hardware components)
Sensor Network Architectures (Concept of sink and source, Topologies, Design Principles)
Radio Propagation and Propagation Impairments
MAC Protocol: Fundamentals, STEM, S-MAC, LEACH, IEEE 802.15.4

UNIT – VI   ROUTING IN SENSOR NETWORKS AND RECENT TRENDS   6 Hours

Routing in MANET : AODV, DSDV, DSR
Naming and Addressing in WSN : Basic concepts, MAC address, Distributed assignment of locally unique
addresses, Content-based and geographic addressing
Routing in Sensor Networks : Challenges and design issues
Routing Protocols for WSNs: Flooding, SPIN, PEGASIS, Directed Diffusion, Geographic Routing
Recent Trends (References web, no formal text)
Software Defined Networking
Wi-FiOffloads
100G Ethernet and its variants
Internet of Things (IoT) and Web of Things
Bring your own Device (BYOD)

Text Books

1. Andrew S. Tanenbaum, "Computer Networks", PHI, Fifth Edition, ISBN : 978-0132-126953
2. KazemSohraby, Daniel Minoli, TaiebZnati, Wireless Sensor Network", Wiley, ISBN :978-0-471-74300-2
3. C. Siva Ram Murthy and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols" Prentice Hall, 2004

Reference Books

1. James F. Kurose and Keith W. Ross, "Computer Networking: A Top-Down Approach Featuring the Internet", Pearson Education, 6th Edition, ISBN : 978-02737-68968
2. Holger Karl and Andreas Willig, "Protocols and Architectures for Wireless Sensor Networks", Wiley, ISBN: 0-470-09510-5
3. Feng Zhao and Leonidas J. Guibas, "Wireless Sensor Networks: An Information Processing Approach" Morgan Kaufmann, 2004
4. Natalia Olifer, Victor Olifer "Computer Network Principles Technologies and Protocols for Network Design", Wiley, ISBN : 978-81-265-0917-1

314442 : THEORY OF COMPUTATION

Teaching Scheme:
Lectures: 4 Hours/Week

Examination Scheme:

In-Semester Assessment	End-Semester Assessment
Phase I – 30 Marks	Phase II – 70 Marks

Prerequisites : Discrete Structure

Course Objectives :
1. To Study computing machines by describing, classifying and comparing different types of computational models.
2. Encourage students to study & develop fundamentals for 'Computational Theory'.

Course Outcomes :
1. Students should be able to understand and design Regular Grammar, Finite Automata, Context Free Grammar, Pushdown Automata, Post Machines, and Turing Machines.
2. Students should be able to Simplify Context Free Grammar and then convert to CNF and GNF.
3. Students should be able to understand Pumping Lemma, Properties of Regular Languages and Context Free Languages.
4. Students should be able to understand Decidable Languages and Turing Reducibility.

UNIT - I   INTRODUCTION   10 Hours
Basic Concepts: Symbols, Strings, Language, Formal Language, Natural Language. Basic Machine And Finite State Machine.

Finite Automata: Definition And Construction - DFA , NFA, NFA with epsilon-Moves, Minimization Of FA, Equivalence of NFA and DFA, Conversion of NFA with epsilon moves to NFA, Conversion of NFA With epsilon moves to DFA, Moore and Mealy Machines, Inter-conversion between Moore and Mealy Machines.

UNIT - II   REGULAR EXPRESSIONS, REGULAR GRAMMAR AND LANGUAGES   8 Hours
Definition and Identities of Regular Expressions, Regular Grammar and Finite Automata: FA to RG and RG to FA, Left Linear and Right Linear Grammar and Inter-conversion between them. Closure Properties of RLs, Pumping Lemma for RL.

UNIT - III   CONTEXT FREE GRAMMAR AND LANGUAGES   8 Hours
Definition and Construction of CFG, Definition and Generation of CFL from CFG. Ambiguous Grammar and Removal of Ambiguity. Simplification of Grammar. Normal Forms of Grammar: CNF and GNF. Chomsky Hierarchy.

UNIT - IV   PUSHDOWN AUTOMATA   8 Hours
Definition and Construction of DPDA and NPDA. Equivalence of PDAs and CFGs, Closure Properties Of CFLs, Concept of Post Machines.

UNIT - V   TURING MACHINES   8 Hours
Definition and Construction of Turing Machines. Languages of TM. Types of TM. Time Complexity of TM, Halting Problem, Church's Turing Hypothesis, Comparison And Applications of DFA, PDA and TM.

UNIT - VI   DECIDABILITY AND REDUCIBILITY   6 Hours
Decidable Languages, Decidable Problems Concerning Regular Languages, Decidable Problems Concerning Context-Free Languages. Decidable Problems With The TM, Turing Reducibility.

Text Books

1. John C. martin, "Introduction to Language and Theory of Computation", TMH, Third Edition. 978-0-07-066048-9.
2. Michel Sipser "Introduction to Theory of Computation" Thomson Course Technology, Second Edition 0-534-95097-3.
3. Kavi Mahesh, "Theory of Computation" Wiley-India, ISBN : 978-81-265-3311-4

Reference Books

1. Hopcroft Ulman, "Introduction To Automata Theory, Languages And Computations", Pearson Education Asia, 2nd Edition
2. Daniel I.A. Cohen, "Introduction to Computer Theory" Wiley-India, ISBN: 978-81-265-1334-5
3. E V Krishnamurthy, "Introduction to Theory of Computer Science", EWP Second 2nd Edition.
4. K.L.P Mishra, N. Chandrasekaran, " Theory Of Computer Science(Automata, Languages and Computation)", Prentice Hall India, 2nd Edition
5. Daniel I.A. Cohen, "Introduction to Automata Theory Languages and Computations", Pearson Education Asia, Second Edition.

314443 : DATABASE MANAGEMENT SYSTEMS

Teaching Scheme:
Lectures: 4 Hours/Week

Examination Scheme:

In-Semester Assessment	End-Semester Assessment
Phase I – 30 Marks	Phase II – 70 Marks

Prerequisites :

Course Objectives :
1. Understand the fundamental concepts of database management. These concepts include aspects of database design, database languages, and database-system implementation.
2. To provide a strong formal foundation in database concepts, technology and practice.
3. To give systematic database design approaches covering conceptual design, logical design and an overview of physical design.
4. Be familiar with the basic issues of transaction processing and concurrency control.
5. To learn and understand various Database Architectures and Applications.
6. Understand how analytics and big data affect various functions now and in the future.

Course Outcomes :
1. Define basic functions of DBMS & RDBMS.
2. Analyze database models & entity relationship models.
3. Design and implement a database schema for a given problem-domain
4. Populate and query a database using SQL DML/DDL commands.
5. Programming PL/SQL including stored procedures, stored functions, cursors and packages
6. Appreciate the impact of analytics and big data on the information industry and the external ecosystem for analytical and data services

UNIT - I   INTRODUCTION   6 Hours
Database Concepts, Database System Architecture, Data Modeling : Data Models, Basic Concepts, entity, attributes, relationships, constraints, keys, E-R and EER diagrams: Components of E-R Model, conventions, converting E-R diagram into tables, EER Model components, converting EER diagram into tables, legacy system model, Relational Model: Basic concepts, Attributes and Domains, Codd's Rules, Relational Integrity: Domain, Entity, Referential Integrities, Enterprise Constraints, Views, Schema Diagram, Database Design : Functional Dependency,
Purpose of Normalization, Data Redundancy and Update Anomalies, Functional Dependency-Single Valued Dependencies. Single Valued Normalization: 1NF, 2NF, 3NF, BCNF. Decomposition: lossless join decomposition and dependency preservation, Multi valued Normalization (4NF), Join Dependencies and the Fifth Normal Form.

UNIT - II   SQL AND PL/SQL   6 Hours
Introduction to SQL: Characteristics and advantages, SQL Data Types and Literals, DDL, DML, DCL, TCL, SQL Operators, Tables: Creating, Modifying, Deleting, Views: Creating, Dropping, Updating using Views, Indexes, Nulls
SQL DML Queries: SELECT Query and clauses, Set Operations, Predicates and Joins, Set membership, Tuple Variables, Set comparison, Ordering of Tuples, Aggregate Functions, Nested Queries, Database Modification using SQL Insert, Update and Delete Queries, concept of Stored Procedures, Cursors, Triggers, assertions, roles and privileges Programmatic SQL: Embedded SQL, Dynamic SQL, Advanced SQL-Programming in MYSQL,SQL 2.0/SQL for OODB, No SQL- MongoDB

UNIT - III   DATABASE TRANSACTIONS   6 Hours

Basic concept of a Transaction, Transaction Management, Properties of Transactions, Concept of Schedule, Serial Schedule, Serializability: Conflict and View, Cascaded Aborts, Recoverable and Non-recoverable Schedules, Concurrency Control: Need, Locking Methods, Deadlocks, Time-stamping Methods, Optimistic Techniques,   Recovery methods : Shadow-Paging and Log-Based Recovery, Checkpoints, Performance Tuning, Query Optimization with respect to No SQL Database

UNIT - IV   ADVANCE AND EMERGING DATABASE CONCEPT   8 Hours

Database Architectures: Centralized and Client-Server Architectures, 2 Tier and 3 Tier Architecture, Introduction to Parallel Databases, Key elements of Parallel Database Processing, Architecture of Parallel Databases, Introduction to Distributed Databases, Architecture of Distributed Databases, Distributed Database Design, Internet Databases, Database Connectivity using Mongo databases

UNIT - V   LARGE SCALE DATA MANAGEMENT   6 Hours

Introduction to Big Data, XML: DTD, XML Schemas, XQuery, XPath, JSON: Overview, Data Types, Objects, Schema, JSON with Java/PHP/Ruby/Python, Hadoop : HDFS, HBase : Overview, HBase Data Model, HBase Region, Hive, SSD

UNIT - VI   DATA WAREHOUSING AND DATA MINING   8 Hours

Teradata RDBMS, Teradata Technology, Data Warehousing : Introduction, Evolution of Data Warehouse, Characteristics, Benefits, Limitation of Data Warehousing, Main Components of Data Warehouse, Conceptual Models, Data Mart, OLAP, Data Mining : Process, Knowledge Discovery, Goals of Data Mining, Data Mining Tasks, Machine learning for Big Data, Business Intelligence, Business analytics.

Emerging Database Technologies: Introduction, Cloud Computing and Data Management, Mobile Databases, Dealing with Massive Datasets-Map Reduce and Hadoop. Introduction to SQLite database, XML databases.

Text Books

1. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", 6thEdition, McGraw Hill Publishers, ISBN 0-07-120413-X
2. S.K.Singh, "Database Systems : Concepts, Design and Application", 2nd Edition, Pearson, 2013, ISBN 978-81-317-6092-5
3. Connally T., Begg C., "Database Systems", 3rd Edition, Pearson Education, 2002, ISBN 81-7808-861-4

Reference Books

1. Kristina Chodorow, Michael Dirolf, "MangoDB: The Definitive Guide" ,O'Reilly Pubications
2. Tom White, "Hadoop: The Definitive Guide", O'Reilly Pubications
3. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", Elsevier
4. Bill Schmarzo, "Big Data:Understanding How Data Powers Big Business", Wiley,ISBN:978-81-265-4545-2
5. Alex Holmes, "Hadoop in Practice", DreamTech Press, ISBN : 978-93-5119-150-6

314444 : SOFTWARE ENGINEERING

Teaching Scheme:
Lectures: 3 Hours/Week

Examination Scheme:

In-Semester Assessment
Phase I – 30 Marks

End-Semester Assessment
Phase II – 70 Marks

Prerequisites :

Course Objectives :
1. To understand the nature of software complexity in various application domains, disciplined way of software development and software lifecycle process models.
2. To introduce principles of agile software development, the SCRUM process and agile practices.
3. To know methods of capturing, specifying, visualizing and analyzing software requirements.
4. To understand concepts and principles of software design and architecture.
5. To understand user-centeredness approach and principles of designing effective user interfaces.
6. To present formal methods, automation and recent trends in software engineering.

Course Outcomes :
1. Identify unique features of various software application domains and classify software applications.
2. Choose and apply appropriate lifecycle model of software development.
3. Describe principles of agile development, discuss the SCRUM process and distinguish agile process model from other process models.
4. Identify user needs and formulate software specifications.
5. Analyze software requirements by applying various modeling techniques.
6. Translate the requirements model into the design model
7. Describe and demonstrate use of software and user-interface design principles.
8. Explain the cleanroom design method.
9. List and classify CASE tools and discuss recent trends and research in software engineering.

UNIT - I   SOFTWARE ENGINEERING PROCESS   6 Hours
Nature of Software – Application domains, web-apps, mobile-apps, cloud computing, product line software
Introduction to Software Engineering – The discipline, layers, the process (guiding principles), the practice (guiding principles) and myths
Process Models – Generic process model, process assessment and improvement, prescriptive models, specialized models, unified process, product and process

UNIT - II   AGILE DEVELOPMENT PROCESS   6 Hours
Agile Development – Agile manifesto, agility and cost of change, agility principles, myth of planned development, toolset for the agile process
Extreme Programming – XP values, process, industrial XP
SCRUM – process flow, scrum roles, scrum cycle description, product backlog, sprint planning meeting, sprint backlog, sprint execution, daily scrum meeting, maintaining sprint backlog and burn-down chart, sprint review and retrospective
Agile Practices - test driven development, refactoring, pair programming, continuous integration, exploratory testing versus scripted testing

UNIT - III   REQUIREMENTS ENGINEERING   6 Hours

Requirements Capturing - requirements engineering (elicitation, specification, validation, negotiation), eliciting requirements, elicitation techniques, developing use cases, building requirements model, negotiating requirements, requirements monitoring, validating requirements, prioritizing requirements (kano diagram) ,

Requirements Analysis – basics, scenario based modeling, UML models, data modeling, data and control flow model, behavioral modeling using state diagrams
Agile Requirements - user stories, 3 Cs of user story, INVEST characteristics

UNIT - IV   SOFTWARE DESIGN   6 Hours

Software Design – definition of design, translating requirements model to design model, design considerations (quality guidelines and attributes), design concepts, design model, design strategies or methods (function-oriented, data-flow-oriented, object-oriented, data-structure-centered, aspect-oriented), design methods classification , design trade-offs,
Software architecture, architectural styles (data-centered, data-flow, call and return, layered, peer-to-peer, publish-subscribe, event-based, client-server), architectural trade-off analysis method (ATAM), domain-specific architectures and product-lines

UNIT - V   USER INTERFACE DESIGN   6 Hours

User Interface – Seeheim model and definition of user interface
User-centeredness in design - dealing with different types of users, collecting user-requirements, building narratives, creating personas and scenarios
Interface design principles–place the user in control, reduce user's memory load, make interface consistent, Shneiderman's 8 Golden Rules
UI Analysis – context of use, user analysis, task analysis
Interface design steps – user interface design process, applying design steps, interface design issues
Usability - characteristics (ISO, Shneiderman, Nielson) , principles (principle of proximity, visibility, visual feedback, visual prominence, mental models and metaphors, consistency, affordance and constraints, confirmation, Hick's law, Fitt's law)

UNIT - VI   FORMAL METHODS, AUTOMATION AND TRENDS IN SOFTWARE   6 Hours
ENGINEERING

Cleanroom Design – cleanroom strategy, process model, black-box, state-box, clear-box specifications, design refinement and verification, cleanroom testing
Software configuration management – SCM basics, SCM repository, SCM process
CASE – taxonomy, tool-kits, workbenches, environments, components of CASE, categories(upper, lower and integrated CASE tools)
Emerging software engineering trends – technology evolution, process trends, collaborative development, model-driven development, test-driven development , challenges of global software development

Text Books
1. Pressman, R. (2010). Software Engineering: A Practitioner's Approach,7th or 8th Ed. Singapore: McGraw Hill. [ISBN -9780071267823]
2. Schwaber, K. and Beedle, M. (2001). Agile Software Development with SCRUM, 1st Ed. New Jersey : Pearson. [ISBN - 9780130676344]
3. Lowdermilk, T. (2013). User-Centered Design, 1st Ed. Sebastopol : O'Reilly Media. ISBN : 9781449359836

Reference Books

1. Vliet, H. (2012). Software Engineering: Principles and Practice, 3rd Ed. New Delhi: Wiley India Pvt Ltd. Ian Sommerville, Peter Sawyer, Requirement Engineering A Good Practice Guide, Wiley India Edition, ISBN : 978-81-265-2457-0
2. Brooks, F. (1995). The Mythical Man Month - Essays on Software Engineering ANV SUB 2nd Edition. Addison Wesley, ISBN - 9780201835953
3. Somerville, I. (2010). Software Engineering, 9th Ed. New Jersey: Pearson Education, ISBN - 9788131762165
4. Mall, R. (2009). Fundamentals of Software Engineering, 3rd Ed. Prentice Hall India, ISBN - 978812033819
5. Jalote, P. (2011). An Integrated Approach to Software Engineering, 3rd Ed. Narosa Publishing House, ISBN - 9788173197024
6. Cohn, M. (2010). Succeeding with Agile: Software Development Using Scrum. Pearson Education, ISBN – 9788131732267
7. Ruby, S. et. Al (2011). Agile Web Development with Rails 4th Ed. Pragmatic, ISBN - 9789350234303 [ISBN – 9788126527373]
8. Murphy, B., Bird, C., Zimmermann, T., Williams, L., Nagappan, N., &Begel, A. (2013, October). Have Agile Techniques been the Silver Bullet for Software Development at Microsoft?. In Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on (pp. 75-84). IEEE.
9. Abrahão, S., Insfran, E., Carsí, J. A., &Genero, M. (2011). Evaluating requirements modeling methods based on user perceptions: A family of experiments. Information Sciences, 181(16), 3356-3378.
10. Massanari, A. L. (2010). Designing for imaginary friends: information architecture, personas and the politics of user-centered design. new media & society, 12(3), 401-416.
11. da Silva, T. S., Martin, A., Maurer, F., &Silveira, M. S. (2011, August). User-Centered Design and Agile Methods: A Systematic Review. In AGILE (pp. 77-86).
12. O'English, M. R. (2011). Applying web analytics to online finding aids: Page views, pathways, and learning about users. Journal of Western Archives, 2(1)

Web Resources

1. http://www.umsl.edu/~sauterv/analysis/F08papers/ View.html#Introduction_8121203202754259
2. http://nptel.ac.in/courses/Webcourse-contents/IISc-BANG/System%20Analysis%20and%20Design/pdf/Lecture_Notes/LNm14.pdf
3. http://www.rspa.com/spi/index.html#webe
4. http://www-itec.uni-klu.ac.at/~harald/proseminar/web11.pdf
5. http://58.59.135.118:8081/BOOKS%5C026%5C21%5CHXYWPJH144310.pdf
6. www.scrum.org
7. http://www.slideshare.net/abhirajoria/web-analytics-and-metrics
8. http://www.slideshare.net/awahid/web-engineering-2337102
9. http://cs.queensu.ca/home/cordy/Papers/ACD_STVR_Survey.pdfhttp://www.digitalanalyticsassociation.org/Files/PDF_standards/ WebAnalyticsDefinitionsVol1.pdf

314445 : WEB ENGINEERING AND TECHNOLOGY

Teaching Scheme:
Lectures: 3 Hours/Week

Examination Scheme:

In-Semester Assessment
Phase I – 30 Marks

End-Semester Assessment
Phase II – 70 Marks

Prerequisites : Basic computer network and Database Management System

Course Objectives :
1. To understand the concepts, principles, strategies, and methodologies of Web applications and development
2. to apply current Web technologies to understand current Web business models
3. to understand and apply Web development processes
4. to understand the engineering aspect of web technology

Course Outcomes :
At the end of this course, students would be able to
1. apply the concepts, principles and methods of Web engineering;
2. have a sufficient theoretical knowledge and analytical skills to develop Web applications;
3. apply the described concepts, principles and methods to development of complex Web applications;
4. design and develop website using current Web technologies and
5. model, visualize and document the analysis and design of Web applications

UNIT – I   WEB ENGINEERING INTRODUCTION   9 Hours
Introduction - What is Web Engineering? Web Development History, Motivation, Categories of Web Applications, Web Applications Characteristics. Evolution and Need for Web Engineering, Web Engineering Models, Software Engineering v/s Web Engineering
World Wide Web: Introduction to TCP/IP and WAP, DNS, Email, TelNet, HTTP and FTP. Introduction to Browser and search engines, Search fundamentals, Search strategies,
Directories search engines and Meta search engines, Working of the search engines ,
Miscellaneous Web Browser details, Introduction to Web Servers: Features of web servers, caching, case study-IIS, Apache, Configuring web servers.

UNIT - II   BASIC MODELS AND ARCHITECTURES   8 Hours
The role of the Information Architect, Collaboration and Communication, Organizing Information, Organizational Challenges, Organizing Web sites parameters and Intranets Creating Cohesive Websites: Conceptual Overview Website Development, Website Design issues, Conceptual Design, High-Level Design, Indexing the Right Stuff, Grouping Content. Architectural Page Mockups, Design Sketches, Navigation Systems. Searching Systems Good & bad web design, Process of Web Publishing. Phases of Web Site development, enhancing your web-site, submission of website to search engines Web security issues, security audit of websites, Web effort estimation, Productivity, Measurement, Quality usability and reliability. Requirements Engineering for Web Applications: Introduction, Fundamentals, Requirement Source, Type, Notations Tools. Principles Requirements Engineering Activities, Adapting RE Methods to Web Application.

UNIT - III   TECHNOLOGIES FOR WEB APPLICATIONS   8 Hours
Introduction HTML: HTML and DHTML, HTML Basic Concepts, Static and dynamic HTML, Structure of HTML documents, HTML Elements, Linking in HTML, Anchor Attributes, Image Maps, Meta Information, Image Preliminaries, Layouts, Backgrounds, Colors and Text, Fonts, Tables, Frames and layers, Audio and

Video Support with HTML Database integration, CSS, Positioning with Style sheets, Forms Control, Form. Elements. Difference between HTML and HTML5

Applying Styles, values, selectors, class, ids, inheritance, layout, backgrounds, borders, margin, padding, lists, fonts, text formatting, positioning, Dynamic HTML, DHTML with CSS.

UNIT - IV   JavaScript &PHP MySQL ADMIN   7 Hours
Introduction to JavaScript: Introduction to Scripting, Difference between Java and JavaScript, JavaScript Characteristics, JavaScript and Common Programming Concepts

Introduction, Benefits Of Using PHP, MySQL, Server/Client Environment, Development Concept: How PHP Script Work, PHP Syntax, Embed PHP In HTML/HTML In PHP, PHP Data Types, Variable In PHP, Operator in PHP, Control Structure, Looping Structure, Function, File Inclusion: Include()/ Require(), Array, String Function: chr()/ strlen()/ strpos()/strcmp(), State Management : Cookies, Session management.

Working with PHP My Admin: Types Data Type, Creating Database & Tables, Dropping Database & Tables, Adding Fields, Selecting Table MySQL Function in PHP : Database Connections, Managing Database Connections, Performing Queries, Closing Connection.

UNIT - V   JAVA SERVLETS   8 Hours
The J2EE and its multitier architecture, The enterprise application, Java Servlet and CGI, A simple servlet, life cycle of servlet, anatomy of servlet, javax.servlet package, Generic servlet, Http Servlet, reading data from client, HTTP request headers, sending data to client, HTTP response header, cookies, session management.

Introduction of XML, Validation of XML documents, DTD, Ways to use XML, XML for data files, HTML Vs XML, Embedding XML into HTML documents, Converting XML to HTML for Display, Displaying XML using CSS and XSL, Rewriting HTML as XML, Relationship between HTML, SGML and XML, web personalization , Semantic web, Semantic Web Services, Ontology.

UNIT - VI   CONTENT MANAGEMENT SYSTEMS   9 Hours
Introduction to CMS, advantages using CMS, CMS development tools: Wordpress, Drupal, Joomla.
Wordpress: content and conversion, directory, file structure, local working, component administration, core, loop, data management, Wordpress as CMS, Wordpress in enterprise.
Website Deployment: Domain registration, Domain hosting, parking websites, uploading data using FTP, email configuration. AJAX

Text Books
1. Steven M. Schafer, "HTML, XHTML and CSS", Fourth Edition by, Wiley India Edition. ISBN: 978-81-265-1635-3.
2. Ivan Bayross, "Web Enabled Commercial Application Development Using HTML, JavaScript, DHTML and PHP", 4th Edition, BPB Publications.
3. Jason Hunter, "Java Servlet Programming", 2nd Edition, O'reilly Publications.

Reference Books

1. Roger S. Pressman, David Lowe, "Web Engineering", Tata Mcgraw Hill Publication, 2007
2. Jim Keogh, "J2EE: The Complete Reference", 1st Edition , Tata McGraw Hill Pulishing Company, ISBN: 978-0-07-052912-0
3. Professional WordPress: Design and Development by Brad Williams, David Damstra, Hal Stern, Wrox publications
4. Ralpha Moseley, M.T. Savaliya, "Developing Web Applications", Wiley, 2nd Edition, ISBN : 978-81-265-3867-6
5. Web Technologies Black Book: HTML, JavaScript, PHP, Java, JSP, XML and AJAX by Kogent Learning Solutions Inc.
6. Gerti Kappel, Birgit Proll, Web Engineering, John Willey and Sons Ltd, ISBN : 978-81-265-2162-3
7. J2EE Architecture, an illustrative gateway to enterprise solutions: concept to Application Design and deployment by B. V. Kumar, S. Sangeetha, S. V. Subrahmanya, Tata McGraw Hill Publishing Company.
8. WordPress for Web developers: An introduction to web professionals by Stephanie Leary, Apress Publications.
9. Dr. Hiren Joshi, Web Technology and Application Development, DreamTech, ISBN : 978-93-5004-088-1

314446 : SOFTWARE LABORATORY – I

Teaching Scheme:
Practical : 4 Hours/Week

Examination Scheme:
Term Work : 50 Marks   Practical : 50 Marks

Prerequisites : Discrete Structure

Course Objectives :
Part-I
1. To design and implement small size network and to understand various networking commands
2. To provide the knowledge of various networking tools and their related concepts
3. To understand various application layer protocols for its implementation in client/server environment

Course Outcomes :
Part-I
1. Students will be implement small size network and its use of various networking commands
2. Students will be able to understand and use various networking tools
3. Students can configure various client/server environments to use application layer protocols

PART – I (Computer Network Technology)
1. Installation of Local Area Network (cables, connectors, topologies, switches/ hubs, crimping tool, IP addressing scheme, Subnetting, College Network Design)
   a. Cross-over cable
   b. Star Topology
   c. Hierarchical Topology
2. Introduction to server administration (server administration commands and their applications) and configuration of
   a. Telnet
   b. FTP
   c. DHCP
3. UNIX Sockets: WAP program in C/C++ sockets API
   a. TCP sockets
   b. UDP sockets

   Server accepts operation and floating point numbers from the clients; performs arithmetic operations and sends the result back to client. Server application must handle at least five clients simultaneously. Both the server and client should display input and output numbers as well as the operation. The server and client processes should be run on different machines. During evaluation, students will demonstrate via creating multiple client processes on different machines.
4. Using a Network Simulator (e.g. packet tracer) configure
   a. Static Routing
   b. RIPv2 routing protocol
   c. EIGRP
   d. OSPF
5. Using a Network Simulator (e.g. packet tracer) configure
   a. VLAN, Dynamic trunk protocol and spanning tree protocol
   b. RIPv2 and EIGRP on same network
   c. WLAN with static IP addressing and DHCP with MAC security and filters

6. Using Network Simulator 2/ OMNET simulate
    a. Local Area Network
    b. WLAN
7. Using Network Simulator 2/ OMNET simulate
    a. MANET
    b. WSN

Infrastructural Requirement:
The laboratories should have the networking components like switches, routers etc.

Term work:
Staff in-charge will suitably frame the above assignments and flexibility may be incorporated. Students will submit term work in the form of journal. Each assignment has to be well documented with problem definition, code documented with comments. Staff in-charge will assess the assignments continuously and grade or mark each assignment on completion date.

Examination:
Oral examination will be based on the term work submitted by the candidate and the associated theory of the assignment performed in the lab.

PART – II (Web Technology)

A. HTML

1. Create following table in HTML

Sr. No.	Name	City	Mobile No.	State Code	State

2. Create a registration form using HTML form input elements viz. textbox, text area, radio button and drop down menu, check box, submit, file and reset button. Field should contain name, address, birth-date, qualification, email, phone number, gender, comments, attach photo etc. Use HTML Form elements wherever required. Align all elements using table.
3. Create a HTML programs using frames. Three frames displaying three different HTML files.
    i. Frame 01 – contain examples of ordered as well as unordered list
    ii. Frame 02 – contain examples of text formatting tags i.e. <b>, <i>, <u>, <h1>--<h6>, <p>, <sup> and <sub>
    iii. Frame 03 – contain example of 3 by 4 table with data.
    iv. Create a hyperlink in Frame 01 that opens in Frame 03

Frame 01	Frame 02
Frame 03	

B. CSS

1. Create a horizontal navigation bar in DIV using external CSS which contain home, about, gallery, enquiry, contacts menus. Also create the same bar in vertical alignment in another DIV in same page.
2. Create a following layout using CSS and display it in html file with proper border, background color, margin, border and padding.
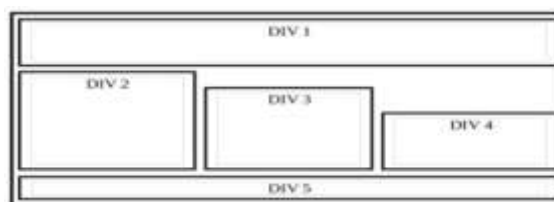
C. PHP

1. Create a PHP program in which two values submitted using form and calculate its addition, subtraction, multiplication, modulation, average and division on the same page. Find the greatest number between them and square of each of them using PHP function.

2. Write following program in PHP:
     i.   Validating given email address
    ii.   Change background color based on hour of a day.
   iii.   Print Fibonacci Series.

D. JavaScript

1. Write a Java script program to create a simple calculator.
2. Write following Java script program:
     i.   Create form validation program that checks the empty values from that form and alert back using alert function. Use at least 5 components.
    ii.   Display a live clock in Java script.

E. PHP MYSQL Admin

1. Create feedback form in PHP which contains first name, last name, address, email, comment and mobile number and store that information in database. Also create a page which displays submitted feedbacks in tabular form.
2. Create a login form using session handling in PHP. After successful login display name, address and other details in tabular format of logged user. Create 5 users. Also create a 'Logout' option. Store the data of user, login and password in the database.
3. Create a simple address book in PHP using MySQL database which contains Adding new person with address (name, phone number, email, permanent address and temporary address etc..), updating their address, deleting him from record and view all records in table.

F. Java Servlets

Create two textboxes on the HTML page named login and password. After clicking on
     i.   the 'login' button the servlet will be displayed. It will show 'login successful' upon correct password else 'authentication failure' will be displayed. Make the use of HTTP Servlet or Generic Servlet.
    ii.   Write a program to demonstrate the use of servlet request and response as well as doGet ( ) and doPost( ) methods. (The subject teacher should provide the appropriate problem statement for this).

G. WordPress

1. Design a website using Content management system of WordPress. Make the use of different plugins and themes of the WordPress (This practical will not be for examination).

Note

All the assignments should be conducted on Latest version of Open Source Operating Systems, tools and Multi-core CPU supporting Virtualization and Multi-Threading.
Subject teacher may frame new assignments which will have equivalent the difficulty level.

The subject teacher should allocate a project of designing a website of any specific organization to the group of two students. The students should use any one client side and one server side technology for the design. Use of content management system is not allowed

Reference Books

1. HTML, XHTML and CSS, Fourth Edition by Steven M. Schafer, Wiley India Edition. ISBN: 978-81-265-1635-3.
2. Web Enabled Commercial Application Development Using HTML, JavaScript, DHTML and PHP, 4th Edition by Ivan Bayross, BPB Publications.
3. J2EE: The Complete Reference, 1st Edition , by Jim Keogh, Tata McGraw Hill Pulishing Company, ISBN: 978-0-07-052912-0
4. Java Servlet Programming 2nd Edition by Jason Hunter, O'reilly Publications.
5. Professional WordPress: Design and Development by Brad Williams, David Damstra, Hal Stern, Wrox publications
6. Web Technologies Black Book: HTML, JavaScript, PHP, Java, JSP, XML and AJAX by Kogent Learning Solutions Inc.
7. J2EE Architecture, an illustrative gateway to enterprise solutions: concept to Application Design and deployment by B. V. Kumar, S. Sangeetha, S. V. Subrahmanya, Tata McGraw Hill Publishing Company.
8. Wordpress for Web developers: An introduction to web professionals by Stephanie Leary, Apress Publications.

314447 : DATABASE MANAGEMENT SYSTEMS LABORATORY

Teaching Scheme:
Practical : 4 Hours/Week

Examination Scheme:

Practical : 50 Marks   Oral : 50 Marks

Prerequisites : Discrete Structure

Course Objectives :
1. To learn and understand Database Programming Paradigms
2. To learn and understand SQL, PL/SQL ,NoSQL
3. To learn Relational Database (Open source) such as MongoDB/ Oracle/MySQL/SQL Server
4. To learn and understand Database Project Life Cycle.

Course Outcomes :
1. Understanding of Database Programming Languages
2. Master the basics of database languages and construct queries using SQL, PLSQL, NoSQL
3. Master the basic concepts of Database Project Life Cycle and appreciate the applications of database systems
4. Understand how analytics and big data affect various functions now and in the future
5. Appreciate the impact of analytics and big data on the information industry and the external ecosystem for analytical and data services

Part – I: Database Programming Languages – SQL, PL/SQL, NoSQL
1. Study of Open Source Databases : MySQL/ MongoDB/CouchDB etc
2. Design and Develop SQL/NoSQL DDL statements which demonstrate the use of SQL objects such as Table, View, Index, Sequence, Synonym.
3. Design at least 10 SQL/NoSQL queries for suitable database application using SQL/NoSQL DML statements: Insert, Select, Update, Delete with operators, functions, and set operator.
4. Design at least 10 SQL/NoSQL queries for suitable database application using SQL DML statements: all types of Join, Sub-Query and View.
5. Write a PL/SQL block to calculate the grade of minimum 10 students.
6. Write a PL/SQL block to implement all types of cursors.
7. Write a PL/SQL stored procedure and function.
8. Write a database Trigger (Row level and Statement level).

Part – II: Large Scale Databases
1. Implement aggregation and indexing with suitable example using MongoDB.
2. Implement Map reduces operation with suitable example using MongoDB.
3. Design and Implement any 5 query using MongoDB
4. Implement any one Concurrency Control Protocol using MongoDB and .net/Java
5. Create simple objects and array objects using JSON
6. Encode and Decode JSON Objects using Java/Perl/PHP/Python/Ruby
7. Implement any machine learning algorithm for BIG data

Part – III : Mini Project : Database Project Life Cycle
1. Design and Implement any Database Application using PHP/ python/Java and MogoDB as a back end. Implement Database navigation operations (add, delete, edit etc.) using ODBC/JDBC.
2. Write a program in PHP/ python/Java/.net to access stored procedure and functions using ODBC/JDBC

- **Requirement Gathering and Scope**
- **Database Analysis and Design**
    - ➢ Design Approach – Bottom Up, Top Down, inside – Out, Mixed strategy.
    - ➢ Design Entity Relationship Model
    - ➢ Relational Model
    - ➢ Database Normalization
- **Implementation :**
    - ➢ Front End : Java/Perl/PHP/Python/Ruby/.net
    - ➢ Backend   : MongoDB
    - ➢ Database Connectivity : ODBC/JDBC
- **Testing : Data Validation**

Group of students should submit the Project Report which will be consist of Title of the Project, Abstract, Introduction, scope, Requirements, Data Modeling features, Data Dictionary, Relational Database Design, Database Normalization, Graphical User Interface, Source Code, Testing document, Conclusion.

Instructor should maintain progress report of mini project throughout the semester from project group and assign marks as a part of the term work

Instructions:

- Instructor should frame minimum six Practical Assignments on **Part I**
- Instructor should frame minimum seven Practical Assignments on **Part II**
- Instructor should frame minimum two Practical Assignments and Develop mini project on **Part III**
- Submission of each Practical Assignment should be in the form of handwritten write-ups/ printout of source code and output
- Instructor should assign a mini project to a group of 3 - 4 students based Part III
- Practical Examination will be based on the all topics covered from Part I, Part II & Part III and questions will be asked to judge understanding of practical performed at the time of practical examination

All the assignments should be conducted on Latest version of Open Source Operating Systems, tools and   Multi-core CPU supporting Virtualization and Multi-Threading.

Reference Books

1. "SQL and PL/SQL for Oracle 10g Black Book", Dr. P. S. Deshpande, DreamTech
2. "SQL, PL/SQL: The Programming Language of Oracle", Ivan Bayross, BPB Publication
3. "Oracle 9i/10g The Complete Reference", Kevin Loney, George Koch, Tata McGraw Hill
4. "Managing and Using MySQL", Reese G., Yarger R., King T., Williams H, 2nd Edition, Shroff Publishers and Distributors Pvt. Ltd., ISBN 81 - 7366 - 465 – X
5. " SQL Server – Black Book", Dalton Patrik, DreamTech Press
6. MongoDB ,O'Reilly Publications
7. Hadoop, O'Reilly Publications
8. Data Mining: Concepts and Techniques by Jiawei Han, Micheline Kamber, Jian Pei, Elsevier
9. http://www.tutorialspoint.com/json/

314448 : EMPLOYABILITY SKILL DEVELOPMENT LABORATORY (SUBJECT 1 - 3)

Teaching Scheme:
Lectures: 1 Hour/Week
Practical : 2 Hours/Week

Examination Scheme:
In-Semester Assessment
Term-work – 50 Marks

Preamble :
Some of the topics or units or some part the unit/topic included in the syllabus should give as self-study to the student/group of students. College will select one subject from the following subjects:

Course Objectives :
1. To raise students' awareness of the need to learn from a wide range of opportunities to develop and practice employability skills that will meet the expectations of the employers.
2. To encourage students in taking responsibility for their personal, professional and academic development.
3. To develop an enterprise and entrepreneurship culture among the students.
4. To increase interaction with employers, resulting in a larger number of employment related activities for student learning.
5. To encourage the students to self-study some of the topics/technologies which may not part of their curriculum but helping them to improve their professional skills

Course Outcomes :
Students will be able to
1. Learn and practice the technologies expected by the employers.
2. Have awareness about recent professional trends and practices followed by the employment market.
3. Have an exposure to entrepreneurship traits.
4. Apply their skills for personal and professional growth.
5. Self-study

SUBJECT-1: CLOUD INFRASTRUCTURE AND SERVICES

Course Overview
The Cloud Infrastructure and Services (CIS) course educates participants about cloud deployment and service models, cloud infrastructure, and the key considerations in migrating to cloud computing. For all definitions of cloud computing, the course has resorted to the U.S. National Institute of Standards and Technology as a guide. The course covers technologies required to build classic (traditional), virtualized, and cloud data center environments. These technologies include compute, storage, networking, desktop and application virtualization. Additional areas of focus include backup/recovery, business continuity, security, and management. Students will learn about the key considerations and steps involved in transitioning from the current state of their data center to a cloud computing environment. Upon completing this course, participants will have the knowledge to make informed decisions about migrating to cloud infrastructure and choosing the best deployment model for their organization.

UNIT – I   JOURNEY TO THE CLOUD   5 Hours
This unit focuses on the business drivers, definition, essential characteristics, and phases of journey to the Cloud. Business drivers for Cloud computing, Definition of Cloud computing, Characteristics of Cloud computing as per NIST, Steps involved in transitioning from Classic data center to Cloud computing environment.

UNIT – II   CLASSIC DATA CENTER (CDC)   5 Hours

This unit focuses on the key elements of CDC – compute, storage, and network, with focus on storage networking, business continuity, and data center management. Application, DBMS, Compute, Storage and Networking, Object based and Unified storage technologies, Business continuity overview and backup, Replication technologies, CDC Management.

UNIT – III   VIRTUALIZED DATA CENTER (VDC)   5 Hours

This unit focuses on virtualization of core technologies in a data center, leading to Virtualized Data Center (VDC). It explains the fundamental concepts of compute, storage, networking, desktop and application virtualization. Concepts and techniques employed for ensuring business continuity in a virtualized data center. Compute, Storage, Network virtualization techniques, Virtual machine (VM) components and process of converting physical to VMs, Block and file level storage virtualization, Virtual provisioning and automated storage tiering, Virtual LAN (VLAN) and Virtual SAN (VSAN) and their benefits, Key network traffic management techniques in VDC, Methods for implementing desktop virtualization, their benefits, and considerations, Application virtualization methods, benefits, and considerations, Backup and recovery of Virtual Machines (VMs), VM replication and migration technologies, Recovery options from total site failure due to a disaster

UNIT – IV   CLOUD COMPUTING AND INFRASTRUCTURE   5 Hours

This unit focuses on the essential characteristics of Cloud Computing, the different Cloud services and deployment models, the economics of Cloud, Cloud infrastructure components, and Cloud service creation processes. Cloud service management processes that ensure that the delivery of Cloud services is aligned with business objectives and expectations of Cloud service consumers. Cloud services models, Cloud deployment models, Economics of Cloud, Cloud infrastructure components, Cloud service creation processes, Cloud service management processes

UNIT – V   CLOUD SECURITY AND MIGRATION TO CLOUD   5 Hours

This unit focuses on security concerns and migration considerations to cloud. Key security concerns and threats and details Cloud model suitable for different categories of users. Security concerns and counter measures in a VDC and Cloud environment, Governance, Risk, and Compliance aspects in Cloud, Cloud security best practices, Cloud models suitable for different categories of users, Considerations for choosing applications suitable for Cloud, Different phases to adopt the Cloud

Open Source Software Provided:
1. VMware Player
2. VMware Hypervisor
3. VMware Converter
4. Open Stack
5. KVM
6. VMware V center Converter
7. VMware Compliance Checker
8. Unified Infrastructure Manager
9. Google App Engine
10. Cloud Foundry

Lab Exercises with the above Open Source software's:

1. Creating a Virtual Machine (VM) on a Guest Operating System.
2. Creating a Virtual Machine as a Bare Metal Hypervisor.
3. Dynamically adding and reducing the resources on Hot VMs.
4. Running Multiple Concurrent operating system using a bare Metal Hypervisor.

5. Creating virtual Switches in the hypervisor.
6. Creating a Virtual Network to allow VM to VM communication.
7. Creating of cloud Services to be hosted as either, Infrastructure as a service
8. Using Platform as a Service to upload any application

Reference Books

1. Cloud Computing: A Practical Approach Author: Anthony T. Velte, Publisher: Tata Mcgraw Hill Education Private Limited(2009), ISBN: 0070683514
2. Cloud Computing For Dummies Author: Halper Fern, Kaufman Marcia, Bloor Robin, Hurwit Judith, and Publisher: Wiley India Pvt. Ltd. (2009), ISBN: 8126524871

SUBJECT-2: DATA SCIENCE AND BIG DATA ANALYTICS

Course Overview

This course provides practical foundation level training that enables immediate and effective participation in big data and other analytics projects. It establishes a baseline of skills that can be further enhanced with additional training and real-world experience. The course provides an introduction to big data and a Data Analytics Lifecycle Process to address business challenges that leverage big data. It provides grounding in basic and advanced analytic methods and an introduction to big data analytics technology and tools, including MapReduce and Hadoop. The course has extensive labs throughout to provide practical opportunities to apply these methods and tools to real-world business challenges and includes a final lab in which students address a big data analytics challenge by applying the concepts taught in the course in the context of the Data Analytics Lifecycle.

UNIT – I   INTRODUCTION TO BIG DATA ANALYTICS   5 Hours

This unit focuses on Big Data Overview, State of the Practice of Analytics, Big Data Analytics in Industry Verticals. It also covers Overview of Data Analytics Lifecycle, Discovery, Data Preparation, Model Planning, Model Building, Communicating Results and Findings, Operationalizing.

UNIT – II   USING R FOR INITIAL ANALYSIS OF THE DATA   5 Hours

This Unit focuses on Introduction to Using R Initial Exploration and Analysis of the Data Using R Basic Data Visualization Using R. Use the R package as a tool to perform basic data analytics, reporting, and apply basic data visualization techniques to your data. Apply basic analytics methods such as distributions, statistical tests and summary operations, and differentiate between results that are statistically sound vs. statistically significant. Identify a model for your data and define the null and alternative hypothesis

UNIT – III   ADVANCED ANALYTICS AND STATISTICAL MODELING FOR BIG DATA –   5 Hours
           THEORY AND METHODS

This unit focuses on examining analytic needs and select an appropriate technique based on business objectives; initial hypotheses; and the data's structure and volume. Apply some of the more commonly used methods in Analytics solutions Explain the algorithms and the technical foundations for the commonly used methods. Explain the environment (use case) in which each technique can provide the most value. Use appropriate diagnostic methods to validate the models created
Use R and in-database analytical functions to fit, score and evaluate models

UNIT – IV   ADVANCED ANALYTICS AND STATISTICAL MODELING FOR BIG DATA –   5 Hours
           TECHNOLOGY & TOOLS

This unit is about learning various tools to Perform Analytics on Unstructured data using MapReduce Programming paradigm. Use Hadoop, HDFS, HIVE,PIG and other products in the Hadoop ecosystem for unstructured data analytics
Effectively use advanced SQL functions and Greenplum extensions for in-database analytics. Use MADlib to solve analytics problems in-database

UNIT – V   ENDGAME - OPERATIONALIZING AN ANALYTICS PROJECT   5 Hours

This unit focuses on Articulate three tasks needed to operationalize an analytics project. Explain how the four common deliverables of an analytics lifecycle project meet the needs of key stakeholders. Use a framework for creating final presentations for sponsors and analysts. Evaluate a data visualization and identify ways to improve it. Apply these concepts to a big data analytics problem in the final lab.

Reference Books
1. Cloud Computing: A Practical Approach Author: Anthony T. Velte, Publisher: Tata Mcgraw Hill Education Private Limited(2009), ISBN: 0070683514
2. Cloud Computing For Dummies Author: Halper Fern, Kaufman Marcia, Bloor Robin, Hurwit Judith, Publisher: Wiley IndiaPvt. Ltd. (2009), ISBN: 8126524871

LABORATORY EXERCISE - DATA SCIENCE AND BIG DATA ANALYTICS

Open Source Software's used:
   a. **R Studio**
   b. **Greenplum Hadoop**

Lab. 1   Introduction to Data Environment

Purpose

The first lab introduces the *Analytics Lab Environment* you will be working on throughout the course. After completing the tasks in this lab you should able to:
   a. Authenticate and access the Virtual Machine (VM) assigned to you for all of your lab exercises
   b. Use SQL and Meta commands in PSQL to navigate through the data sets
   c. Create subsets of the *data*, using *table joins and filters* to analyze subsequent lab exercises

Tasks

Tasks you will complete in this lab exercise include:
   a. Exploring databases and datasets
   b. Using PSQL statements and Meta commands.
   c. Creating subsets of data for use in subsequent lab exercises

Lab. Reference
   a. PSQL Commands – Quick Reference
   b. PSQL Meta Commands – Quick Reference
   c. Surviving LINUX – Quick Reference
   d. R – Quick Reference

Lab. 2   Introduction to R

Purpose

This lab introduces you to the use of the R statistical package within the Data Science and Big Data Analytics environment. After completing the tasks in this lab you should able to:
   a. Read data sets into R, save them, and examine the contents

Tasks

Tasks you will complete in this lab include:
   a. Invoke the R environment and examine the R workspace
   b. Read tables created in Lab 1 into the R statistical package
   c. Examine, manipulate and save data sets
   d. Exit the R environment

Lab. Reference
   a. R Commands – Quick Reference
   b. Surviving LINUX – Quick Reference

Lab. 3   Basic Statistics, Visualization, and Hypothesis Tests

Purpose

The lab introduces you to the analysis of data using the R statistical package within the Data Science and Big Data Analytics environment. After completing the tasks in this lab you should able to:

    a.   Perform summary (descriptive) statistics on the data sets

    b.   Create basic visualizations using R both to support investigation of the data as well as exploration of the data

    c.   Create plot visualizations of the data using a graphics package

    d.   Test a hypothesis about the data

Tasks

Tasks you will complete in this lab include:

    a.   Reload data sets into the R statistical package

    b.   Perform summary statistics on the data

    c.   Remove outliers from the data

    d.   Plot the data using R

    e.   Plot the data using lattice and ggplot

    f.   Test a hypothesis about the data

Lab. Reference

    a.   R Commands - Quick Reference

    b.   Surviving LINUX - Quick Reference

Lab. 4   K-means Clustering

Purpose

This lab is designed to investigate and practice K-means Clustering. After completing the tasks in this lab you should able to:

    a.   Use R functions to create K-means Clustering models

    b.   Use ODBC connection to the database and execute SQL statements and load datasets from the database in an R environment

    c.   Visualize the effectiveness of the K-means Clustering algorithm using graphic capabilities in R

    d.   Use MADlib functions for K-means clustering

Task

Tasks you will complete in this lab include:

    a.   Use the R -Studio environment to code K-means Clustering models

    b.   Use the ODBC connection in the R environment to create the average household income from the census database as test data for K-means Clustering

    c.   Use R graphics functions to visualize the effectiveness of the K-means Clustering algorithm

    d.   Use MADlib functions for K-means clustering

Lab. Reference

    http://www.statmethods.net/advstats/cluster.html (originally from Everitt & Hothorn).

Lab. 5   Association Rules

Purpose

This lab is designed to investigate and practice Association Rules. After completing the tasks in this lab you should able to:
   a. Use R functions for Association Rule based models

Tasks

Tasks you will complete in this lab include:
   a. Use the R -Studio environment to code Association Rule models
   b. Apply constraints in the Market Basket Analysis methods such as minimum thresholds on support and confidence measures that can be used to select interesting rules from the set of all possible rules
   c. Use R graphics "arules" to execute and inspect the models and the effect of the various thresholds

Lab. Reference

The groceries data set - provided for arules by Michael Hahsler, Kurt Hornik and Thomas Reutterer.
http://rss.acs.unt.edu/Rdoc/library/arules/html/Groceries.html
   a. Michael Hahsler, Kurt Hornik, and Thomas Reutterer (2006) Implications of probabilistic data modeling for mining association rules. In M. Spiliopoulou, R. Kruse, C. Borgelt, A. Nuernberger, and W. Gaul, editors, From Data and Information Analysis to Knowledge Engineering, Studies in Classification, Data Analysis, and Knowledge Organization, pages 598-605. Springer-Verlag.

Lab. 6   Linear Regression

Purpose

This lab is designed to investigate and practice the Linear Regression method. After completing the tasks in this lab you should able to:
   a. Use R functions for Linear Regression (Ordinary Least Squares - OLS)
   b. Predict the dependent variables based on the model
   c. Investigate different statistical parameter tests that measure the effectiveness of the model

Task

Tasks you will complete in this lab include:
   a. Use the R -Studio environment to code OLS models
   b. Review the methodology to validate the model and predict the dependent variable for a set of given independent variables
   c. Use R graphics functions to visualize the results generated with the model

Lab. Reference

   a. R Commands - Quick Reference
   b. Surviving LINUX - Quick Reference

Lab. 7   Logistic Regression

Purpose

This lab is designed to investigate and practice the Logistic Regression method. After completing the tasks in this lab you should able to:
   a. Use R functions for Logistic Regression - also known as Logit)
   b. Predict the dependent variables based on the model
   c. Investigate different statistical parameter tests that measure the effectiveness of the model

Tasks

Tasks you will complete in this lab include:
a. Use R -Studio environment to code Logit models
b. Review the methodology to validate the model and predict the dependent variable for a set of given independent variables
c. Use R graphics functions to visualize the results generated with the model

Lab. Reference
a. R Commands - Quick Reference
b. Surviving LINUX - Quick Reference

Lab. 8   Naive Bayesian Classifier

Purpose

This lab is designed to investigate and practice the Naïve Bayesian Classifier analytic technique. After completing the tasks in this lab you should be able to:
a. Use R functions for Naïve Bayesian Classification
b. Apply the requirements for generating appropriate training data
c. Validate the effectiveness of the Naïve Bayesian Classifier with the big data

Task

Tasks you will complete in this lab include:
a. Use R -Studio environment to code the Naïve Bayesian Classifier
b. Use the ODBC connection to the "census" database to create a training data set for Naïve Bayesian Classifier from the big data
c. Use the Naïve Bayesian Classifier program and evaluate how well it predicts the results using the training data and then compare the results with original data

Lab. Reference
a. R Commands - Quick Reference
b. Surviving LINUX - Quick Reference

Lab. 9   Decision Trees

Purpose

This lab is designed to investigate and practice Decision Tree (DT) models covered in the course work. After completing the tasks in this lab you should able to:
a. Use R functions for Decision Tree models
b. Predict the outcome of an attribute based on the model

Tasks

Tasks you will complete in this lab include:
a. Use the R -Studio environment to code Decision Tree Models
b. Build a Decision Tree Model based on data whose schema is composed of attributes
c. Predict the outcome of one attribute based on the model

Lab. Reference
a. R Commands - Quick Reference
b. Surviving LINUX - Quick Reference.

Lab. 10   Time Series Analysis with ARIMA

Purpose

This lab is designed to investigate and practice Time Series Analysis with ARIMA models (Box-Jenkins-methodology). After completing the tasks in this lab you should able to:
    a.   Use R functions for ARIMA models
    b.   Apply the requirements for generating appropriate training data
    c.   Validate the effectiveness of the ARIMA models

Task

Tasks you will complete in this lab include:
    a.   Use the R -Studio environment to code ARIMA models
    b.   Use the ODBC connection to the database to create the weekly sales data from the retail database
    c.   Prepare the data (sorting and rendering the data as a Time series)
    d.   Generate a model and evaluate how well it predicts the results and compare the results with original data

Lab. Reference
    a.   R Commands - Quick Reference
    b.   Surviving LINUX - Quick Reference

Lab. 11   Hadoop, HDFS, MapReduce and Pig

Purpose

This lab introduces the Hadoop and MapReduce environment that you will be working on for the next lab. After completing the tasks in this lab you should able to:
    a.   Get help on the various Hadoop commands
    b.   Observe a MapReduce job in action
    c.   Query various Hadoop servers regarding status
    d.   Understand and execute "Pig" statements

Tasks

Tasks you will complete in this lab include:
    a.   Run Hadoop and Hadoop fs and collect help information
    b.   Run a shell script to perform a word count activity
    c.   Run a MapReduce job to produce similar output
    d.   Investigate the UI for MapReduce/HDFS components to track system behavior
    e.   Run "Pig" statements to execute the same tasks done with MapReduce

Lab. Reference

References used in this lab are located in your Student Resource Guide. See the Guide for:
    a.   Hadoop Commands
    b.   HDFS Commands

Lab. 12   In-database Analytics

Purpose

This lab is designed to familiarize you and give you practice with the in-database analytics methods covered in lessons three and four of Module 5. After completing the tasks in this lab you should able to:
    a.   Use window functions
    b.   Implement user defined aggregates and user defined functions
    c.   Use ordered aggregates

    d.   Use Regular Expressions (Regex) in SQL for text filtering

    e.   Use MADlib functions and plot results from MADlib function outputs

Task

Tasks you will complete in this lab include:

    a.   Process Clickstream analysis data using window functions, User defined functions, User defined aggregates and regular expressions

    b.   Compute median household income using ordered aggregates

    c.   Use MADlib functions for logistic regression and direct output to plot the results

Lab. Reference

Student resource guide

    http://doc.madlib.net/v0.2beta/group__grp__logreg.html

Lab. 13   Final Lab Exercise on Big Data Analytics

Purpose

This lab allows students to apply what they have learned from the analytical methods and tools to a big data problem using the Analytics Lab Environment.

Task

Tasks you will complete in this lab include:

    a.   Explore the big data set provided and prepare the data for analysis

    b.   Assess data quality, outliers and training sets

    c.   Conduct model selection, code, execute and score the model

    d.   Use R and PSQL statements during your analysis of big data

    e.   Create a narrative summary of your findings, using the methods covered earlier in this module

Lab. Reference

References used throughout the labs are located in your Student Resource Guide Appendix. See the Appendix for:

    http://www.ffiec.gov/hmda/

The directory /home/gpadmin/FINAL_LAB in your lab environment will be your working directory for the final lab exercise. Following files are pre-loaded in this lab:

    Analyst.ppt – Analyst presentation template

    Sponsor.ppt – Sponsor presentation template

    *.asc – encrypted files with suggested code for the solution. The decrypting of these files are performed with the following command at the $ prompt in the FINAL_LAB directory:

    gpg -o *.* -d *.*.asc

    (*.* represents the filename with extension name)

You will be prompted for a passphrase. Your instructor will provide the pass phrase

SUBJECT-3: INFORMATION STORAGE & MANAGEMENT

Course Overview

Information Storage and Management (ISM) is the only course of its kind to fill the knowledge gap in understanding varied components of modern information storage infrastructure, including virtual environments. It provides comprehensive learning of storage technology, which will enable you to make more informed decisions in an increasingly complex IT environment. ISM builds a strong understanding of underlying storage technologies and prepares you to learn advanced concepts, technologies, and products. You will learn about the architectures, features, and benefits of Intelligent Storage Systems; storage networking technologies such as FC-SAN, IP-SAN, NAS, Object-based and unified storage; business continuity solutions such as backup, replication, and archive; the increasingly critical area of information security; and the emerging field of cloud computing.

UNIT - I   STORAGE SYSTEM   5 Hours

Introduction to evolution of storage architecture, key data center elements, virtualization, and cloud computing. Key data center elements – Host (or compute), connectivity, storage, and application in both classic and virtual environments. RAID implementations, techniques, and levels along with the impact of RAID on application performance. Components of intelligent storage systems and virtual storage provisioning and intelligent storage system implementations.

UNIT - II   STORAGE NETWORKING TECHNOLOGIES AND VIRTUALIZATION   5 Hours

Fibre Channel SAN components, connectivity options, and topologies including access protection mechanism 'zoning", FC protocol stack, addressing and operations, SAN-based virtualization and VSAN technology, iSCSI and FCIP protocols for storage access over IP network, Converged protocol FCoE and its components, Network Attached Storage (NAS) - components, protocol and operations, File level storage virtualization, Object based storage and unified storage platform.

UNIT - III   BACKUP, ARCHIVE, AND REPLICATION   5 Hours

This unit focuses on information availability and business continuity solutions in both virtualized and non-virtualized environments. Business continuity terminologies, planning and solutions, Clustering and multi-pathing architecture to avoid single points of failure, Backup and recovery - methods, targets and topologies, Data deduplication and backup in virtualized environment, Fixed content and data archive, Local replication in classic and virtual environments, Remote replication in classic and virtual environments, Three-site remote replication and continuous data protection

UNIT - IV   CLOUD COMPUTING CHARACTERISTICS AND BENEFITS   5 Hours

This unit focuses on the business drivers, definition, essential characteristics, and phases of journey to the Cloud. ,Business drivers for Cloud computing, Definition of Cloud computing, Characteristics of Cloud computing, Steps involved in transitioning from Classic data center to Cloud computing environment Services and deployment models, Cloud infrastructure components, Cloud migration considerations

UNIT - V   SECURING AND MANAGING STORAGE INFRASTRUCTURE   5 Hours

This chapter focuses on framework and domains of storage security along with covering security implementation at storage networking. Security threats, and countermeasures in various domains Security solutions for FC-SAN, IP-SAN and NAS environments, Security in virtualized and cloud environments, Monitoring and managing various information infrastructure components in classic and virtual environments, Information lifecycle management (ILM) and storage tiering, Cloud service management activities

Open Source Software : Unisphere

Lab Exercises with Unisphere Includes :
1. Data center System Health
    a. Provide an overview of the VNXe Simulator features and functions
2. Creating a common share
    a. In this lab you'll learn how to create common shared folders using the VNXe storage system and EMC Unisphere tool.
3. Creating aniSCSIgenericstorage
    a. In this lab you'll learn how to create iSCSI generic storages
4. Creating aniSCSItarget
    a. In a data center or enterprise environment an iSCSI target often resides in a large storage array such as EMC VNXe.
    b. To understand the process of an iSCSI target creation you have to know the module 2,4,6,11 of the ISM v2 course:
5. Creating an Exchange Storage Capacity
    a. In this lab you'll learn how to create Microsoft Exchange storage
6. Creating a VMware Storage Capacity
    a. In this lab you'll learn how to create VMware Storage Capacity
7. Creating a Hyper V Storage Capacity
    a. In this lab you'll learn how to create Hyper V Storage Capacity
8. Creating Replication Sessions
    a. In this lab you'll learn how to create Replication Sessions.

Reference Books
1. Information Storage and Management, Author :EMC Education Services, Publisher: Wiley ISBN: 9781118094839
2. Storage Virtualization, Author: Clark Tom, Publisher: Addison Wesley Publishing Company ISBN : 9780321262516

SEMESTER - II

314449 : DESIGN AND ANALYSIS OF ALGORITHMS

Teaching Scheme:
Lectures: 4 Hours/Week

Examination Scheme:

In-Semester Assessment   End-Semester Assessment
Phase I – 30 Marks   Phase II – 70 Marks

Prerequisites : Data Structures, Basic mathematics: Induction, probability theory and Basic searching and sorting algorithms

Course Objectives :
1. To know the basics of computational complexity analysis and various algorithm design paradigms.
2. Provide students with solid foundations to deal with a wide variety of computational problems.
3. To provide a thorough knowledge of the most common algorithms and data structures.
4. To analyze a problem and identify the computing requirements appropriate for its solutions

Course Outcomes :
1. Apply Knowledge of Mathematics to perform asymptotic analysis of algorithms.
2. Demonstrate a familiarity with major algorithms and data structures.
3. Apply important algorithmic design paradigms and methods of analysis.
4. Synthesize efficient algorithms in common engineering design situations.

UNIT - I   INTRODUCTION   **8 Hours**
Analysis of Algorithm, Efficiency- Analysis framework, asymptotic notations – big O, theta and omega. Analysis of Non-recursive and recursive algorithms, Amortized Analysis. Solving Recurrence Equations (Homogeneous and non-homogeneous) Proof Techniques: Minimum 2 examples of each: Contradiction, Mathematical Induction – Tiling Problem, Direct proofs, Proof by counterexample, Proof by contraposition.

UNIT - II   DIVIDE AND CONQUER AND GREEDY   **10 Hours**
Divide & Conquer: General method, Control abstraction, Merge sort, Quick Sort – Worst, Best and average case. Binary search, Large integer Multiplication, Strassen's Matrix multiplication. (for all above algorithms analysis to be done with recurrence)
Greedy Method: General method and characteristics, Prim's method for MST , Kruskal method for MST (using nlogn complexity), Dijkstra's Algorithm, Huffman Trees (nlogn complexity), Fraction Knapsack problem, Job Sequencing

UNIT - III   DYNAMIC PROGRAMMING   **8 Hours**
General strategy, Principle of optimality, Warshal's and Floyd's Algorithm , Optimal Binary Search Trees, 0/1 knapsack Problem, Travelling Salesman Problem

UNIT - IV   BACKTRACKING   **6 Hours**
General method, Recursive backtracking algorithm, iterative backtracking method. 8- queens problem, Sum of subsets, Graph coloring, Hamiltonian Cycle , 0/1 Knapsack Problem

UNIT - V   BRANCH AND BOUND   **8 Hours**
The method, Control abstractions for Least Cost Search, Bounding, FIFO branch and bound, LC branch and bound, 0/1 Knapsack problem – LC branch and bound and FIFO branch and bound solution, Traveling sales person problem

UNIT - VI   COMPUTATIONAL COMPLEXITY AND PARALLEL ALGORITHMS   6 Hours

Non Deterministic algorithms, The classes P, NP, NP Complete, NP hard
Proofs for NP Complete Problems: Clique, Vertex Cover
Parallel Algorithms: Introduction, models for parallel computing, computing with complete binary tree,
Pointer doubling algorithm

Text Books

1. Horowitz and Sahani, "Fundamentals of computer Algorithms", Galgotia. ISBN 81-7371-612-9
2. R.C.T.Lee, S S Tseng, R C Chang, Y T Tsai " Introduction to Design and Analysis of Algorithms, A Strategic approach" Tata McGraw Hill. ISBN-13:978-1-25-902582-2. ISBN-10:1-25-902582-9
3. Gilles Brassard, Paul Bratle "Fundamentals of Algorithms ", Pearson ISBN 978-81-317-1244-3

Reference Books

1. Thomas H Cormen and Charles E.L Leiserson, "Introduction to Algorithm" PHI, ISBN:81-203-2141-3
2. Anany Levitin,"Introduction to the Design & Analysis of Algorithm ",Pearson ISBN 81- 7758-835-4
3. Steven S Skiena, The Algorithm Design Manual, Springer,2nd edition, ISBN 978-81-8489-865-1
4. George T. Heineman, Gary Pollice, Stanley Selkow "Algorithms in a Nutshell, A Desktop Quick Reference", O'Reilly, ISBN 13:978-81-8404-608-3

314450 : SYSTEMS PROGRAMMING

Teaching Scheme:
Lectures: 4 Hours/Week

Examination Scheme:
In-Semester Assessment
Phase I – 30 Marks

End-Semester Assessment
Phase II – 70 Marks

Prerequisites : Data Structures, Computer Organization, Microprocessors, Basic searching and sorting algorithms

Course Objectives :
1. To learn and understand fundamentals of System Software Programs as Assembler, Macroprocessor, Linkers and Loaders.
2. To learn how to design and develop various System Software Programs
3. To study phases of compiler in detail.

Course Outcomes :
At the end of this course, students would be able to
1. design & implement System Programs as Assembler, Macroprocessor.
2. use tool Lex for generation of Lexical Analyzer.
3. use tool YACC for generation of Syntax Analyzer.

UNIT - I   INTRODUCTION TO SYSTEMS PROGRAMMING   12 Hours
Introduction: Components of System Software, Language Processing Activities, Fundamentals of Language Processing
Assemblers: Elements of Assembly Language Programming
A simple Assembly Scheme, Pass structure of Assemblers, Design of Two Pass Assembler, Single pass assembler
Macro Processor: Macro Definition and call, Macro Expansion, Nested Macro Calls and definition, Advanced Macro Facilities, Design of Macro Processor

UNIT - II   LOADERS AND LINKERS   6 Hours
Loaders: Loader Schemes, Compile and Go, General Loader Scheme, Absolute Loader Scheme, Subroutine Linkages, Relocation and linking concepts, Self-relocating programs, Relocating Loaders, Direct Linking Loaders, Overlay Structure, Design of absolute and direct linking loader.

UNIT - III   INTRODUCTION TO COMPILER   8 Hours
Phase structure of Compiler
Lexical Analyzer: The Role of the Lexical Analyzer, Input Buffering. Specification of Tokens, Recognition of Tokens, Lexical Analyzer, RE to DFA Conversion, Lexical Errors.
LEX: LEX Specification, Generation of Lexical Analyzer by LEX.

UNIT - IV   PARSERS   8 Hours
Role of parsers, Top down parsers, recursive descent parser, predictive parser, LL(K) parsers, Bottom up Parsers - Operator Precedence Parser, Shift Reduce - SLR, LR(K), LALR, Error Detection and Recovery in Parser, YACC specification and features Automatic construction of Parser(YACC).

UNIT - V   SEMANTIC ANALYSIS   8 Hours
Need, Syntax Directed Translation, Type Checking, Type conversion, Syntax Directed Definitions, Translation of assignment Statements, iterative statements, Boolean expressions, and conditional statements.

Intermediate Code Formats: Postfix notation, Parse and syntax tress, Three address code, quadruples and triples. Quadruple generation for 1D and 2D arrays

UNIT - VI   STORAGE ALLOCATION, CODE GENERATION AND OPTIMIZATION   8 Hours
Storage allocation, code generation and Code optimization:
Storage Allocation: Run time storage organization, storage organization and allocation strategies.
Code Generation: Code generation Issues, A Simple Code Generator
Code Optimization: Machine Independent: Common Sub-expression elimination, Removing of loop invariants, Reduction in strengths.
Machine dependent Issues: Assignment and use of registers, Rearrangement of Quadruples for code optimization.

Text Books
1. D.M. Dhamdhere ,"Systems Programming and Operating Systems", Tata McGraw-Hill, ISBN-13:978-0-07-463579-7
2. Alfred V. Aho, Ravi Sethi, Reffrey D. Ullman, "Compilers Principles, Techniques, and Tools", Addison Wesley, ISBN 981-235-885-4
3. John J Donovan ,"Systems Programming", Tata McGraw-Hill Edition 1991, ISBN 0-07-460482-1

Reference Books
1. Leland L. Beck, "System Software An Introduction to Systems Programming" 3rd Edition, Person Education, ISBN 81-7808-036-2
2. Adam Hoover,"System Programming with C and Unix", Pearson,2010
3. Terence Parr, "Language Implementation Patterns",SPD,2009

314451 : OPERATING SYSTEM

Teaching Scheme:
Lectures: 4 Hours/Week

Examination Scheme:

In-Semester Assessment	End-Semester Assessment
Phase I – 30 Marks	Phase II – 70 Marks

Prerequisites : Computer Organization and Data Structure

Course Objectives :
1. To Introduce basic concepts and functions of modern operating systems
2. To Understand the concept of process, and thread management
3. To Understand how the resources are scheduled and managed
4. To Understand the concepts of process synchronization and deadlock
5. To know the concept of I/O and File management
6. To Understand various Memory management techniques
7. To be aware of latest trends in Operating Systems

Course Outcomes :
At the end of this course, students would be able to
1. Possess knowledge of the role of Operating Systems and their types.
2. Apply the concept of a process, thread and scheduling algorithms.
3. Apply the concepts of process synchronization and how it is achieved.
4. Realize the concept of deadlock and different ways to handle it.
5. Realize various memory management techniques.
6. Realize the concept of I/O management and File system

UNIT – I   OPERATING SYSTEM OVERVIEW   8 Hours
Operating System Objectives and Functions, The Evolution of Operating Systems, Developments Leading to Modern Operating Systems, Virtual Machines
OS Design Considerations for Multiprocessor and Multicore architectures, Microsoft Windows Overview, Modern UNIX Systems, Linux, Android. Booting Process of all the above operating systems.

UNIT - II   PROCESS DESCRIPTION AND CONTROL   10 Hours
Process: Concept of a Process, Process States, Process Description, Process Control (Process creation, Waiting for the process/processes, Loading programs into processes and Process Termination), Execution of the Operating System.
Threads: Processes and Threads, Concept of Multithreading, Types of Threads, Thread programming Using pthreads, Multicore processors and threads , Linux Process and Thread Management, Android Process and Thread Management
Scheduling: Uniprocessor Scheduling - Types of Scheduling, Scheduling Algorithms, and Thread Scheduling, An introduction to Multiprocessor and Real-Time Scheduling, Traditional UNIX Scheduling, Linux Scheduling.

UNIT - III   CONCURRENCY: MUTUAL EXCLUSION AND SYNCHRONIZATION   8 Hours
Concurrency: Process/thread Synchronization and Mutual Exclusion
Principles of Concurrency, Requirements for Mutual Exclusion, Mutual Exclusion: Hardware Support, Operating System Support (Semaphores and Mutex), Programming Language Support (Monitors), Classical synchronization problems: Readers/Writers Problem, Producer and Consumer problem.
Concurrency : Deadlock and Starvation

Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock detection, An Integrated Deadlock Strategy, Example: Dining Philosophers Problem, Linux inter-process communication and concurrency mechanisms, Android Inter-process communication mechanisms and concurrency mechanisms

UNIT - IV   MEMORY MANAGEMENT   8 Hours
Memory Management: Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Buddy System, Relocation, Paging, Segmentation.
Virtual Memory: Hardware and Control Structures, Operating System Software, Linux Memory Management, Windows Memory Management, Android Memory Management.

UNIT – V   INPUT/OUTPUT AND FILES   8 Hours
I/O Management and Disk Scheduling: I/O Devices, Organization of the I/O Function, Operating System Design Issues, I/O Buffering, Disk Scheduling, Disk Cache, Linux I/O.
File Management: Overview, File Organization and Access, File Directories, File Sharing, Record Blocking, Secondary Storage Management, Linux Virtual File System, Android File Management.

UNIT - VI   RECENT AND FUTURE TRENDS IN OS   6 Hours
Linux Kernel Module Programming, Embedded Operating Systems: Characteristics of Embedded Systems, Embedded Linux, and Application specific OS. Basic services of NACH Operating System. Introduction to Service Oriented Operating System (SOOS), Introduction to Ubuntu EDGE OS, etc.

Text Books
1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition, 2014, ISBN-10: 0133805913 • ISBN-13: 9780133805918
2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, John Wiley & Sons ,Inc., 9th Edition,2012, ISBN 978-1-118-06333-0
3. Maurice J. Bach, "Design of UNIX Operating System", PHI

Reference Books
1. Tom Adelstein and Bill Lubanovic, Linux System Administration, O'Reilly Media, Inc., 1st Edition, 2007.ISBN-10: 0596009526 | ISBN-13: 978-0596009526
2. Harvey M. Deitel, Operating Systems, Prentice Hall, 3rd Edition,2003, ISBN-10: 0131828274 | ISBN-13: 978-0131828278
3. Andrew S. Tanenbaum, Modern Operating System, Prentice Hall, 3rd Edition, 2007,ISBN-10: 0136006639 | ISBN-13: 978-0136006633
4. Operating System in depth by Thomson

314452 : MULTIMEDIA TECHNOLOGIES

Teaching Scheme:
Lectures: 3 Hours/Week

Examination Scheme:

In-Semester Assessment	End-Semester Assessment
Phase I – 30 Marks	Phase II – 70 Marks

Prerequisites : Data Structures and Files and Basics of computer graphics and animation

Course Objectives :
1. To learn 5 basic components of multimedia (text, image, audio, video and animation)
2. To learn the advance graphics
3. To learn compression techniques for various multimedia components
4. To learn Gaming and animation

Course Outcomes :
At the end of this course students will be able to
1. to create their own file formats for specific application
2. to do some projects based on current trends in multimedia
3. To use of open sources for authoring tool for animation and presentations
4. To develop simple games and animation

UNIT – I   INTRODUCTION TO MULTIMEDIA   6 Hours
Goals, objectives, and characteristics of multimedia, Multimedia building blocks, Multimedia architecture, hardware support

Distributed multimedia applications, streaming technologies, multimedia
database systems

Multimedia authoring tools, overview of multimedia software tools, Multimedia Applications Media Entertainment, Media consumption, web-based applications, e-learning and education

Text: Types of text, Text compression: Huffman coding, LZ & LZW
Text file formats: TXT, DOC; RTF, PDF, PS

UNIT - II   DIGITAL IMAGE   6 Hours
Basic Image fundamentals, image File formats - (BMP, TIFF, JPEG, GIF) Image acquisition, storage processing, Communication, and display
Image Enhancement: Enhancement by point processing, Spatial filtering
Image Compression: Types of Compression: Lossy & Lossless, Symmetrical & Asymmetrical, Intra-frame &Inter-frame Hybrid JPEG, Lossless: RLE, Shannon - Fano algorithm, Arithmetic coding.
Lossy: Vector quantization, Fractal Compression Technique, Transform Coding, Psycho-analysis, and inter-frame Correlation. Hybrid: JPEG-DCT

UNIT - III   AUDIO AND AUDIO COMPRESSION   5 Hours
Nature of sound waves, characteristics of sound waves, psycho-acoustic, MIDI, digital audio, CD formats.
Audio file formats: WAV, AIFF, VOC, AVI, MPEG Audio File formats, RMF, WMA
Audio compression techniques : DM, ADPCM and MPEG
Audio file conversions
Multimedia Supported audio formats in Android, Media Playback.

UNIT - IV   VIDEO   5 Hours

Video signal formats, Video transmission standards: EDTV, CCIR, CIF, SIF, HDTV, digitization of video
Video file formats: MOV, Real Video, H-261, H-263, Cinepack, Nerodigtal,Video editing, DVD formats, MPEG, Video streaming
Multimedia Supported video formats in Android, Media Playback.

UNIT - V   ANIMATION AND OPENGL   5 Hours

Animation: Basics of animation, types of animation, principles of animation, techniques of animation, Creating animation
OpenGL: Open GL over windows/Linux, Extension, programming languages, SDK, shadowing techniques, rendering,

UNIT - VI   ADVANCES IN MULTIMEDIA   5 Hours

Virtual Reality : Concept, Forms of VR, VR applications, VR devices: Hand Gloves, Head mounted tracking system, VR chair, CCD, VCR, 3D Sound system, Head mounted display
Synchronization: Multimedia Communication and applications, Study of Multimedia networking, Quality of data transmission, Multimedia over IP, Media on Demand
Multimedia in Android: Android Multimedia Framework Architecture, GStreamer :Introduction, GStreamer Based Multimedia Framework, Open Core MultimediaEngine

Text Books
1. Ralf Steinmetz and Klara Nahrstedt "Multimedia Computing, Communication and Applications", Pearson Education.
2. K.R. Rao, "Multimedia Communication Systems: Techniques, Standards, and Networks", TMH.
3. Ranjan Parekh, "Principles of Multimedia", 2/E, Tata McGraw-Hill, ISBN: 1259006506

Reference Books
1. Ashok Banerji, Ananda Ghosh, "Multimedia Technologies", ISBN: 9780070669239
2. Gonzalez, Woods, "Digital Image Processing" Addison Wesley
3. Ze-Nian Li, Marks S. Drew, "Fundamentals of Multimedia", Pearson Education.
4. Edward Angel, "OpenGL: A Primer", Addison-Wesley.
5. Parag Havaldar, Gerard Medioni, "Multimedia Systems", Cengage Learning.

314453 : INFORMATION TECHNOLOGY PROJECT MANAGEMENT

Teaching Scheme:
Lectures : 3 Hours / Week

Examination Scheme:

In-Semester Assessment
Phase I – 30 Marks

End-Semester Assessment
Phase II – 70 Marks

Prerequisites :

Course Objectives :
1. Explain the need for Engineers in Management
2. To understand basic managerial and behavioral traits essential in industry.
3. To understand project management through life cycle of the project.
4. To understand project planning, execution, tracking, audit and closure of project.
5. To understand basics of ERP systems and its deployment.
6. To understand current and future trends and practices in the IT industry.

Course Outcomes :
1. Students will learn and understand basic essential managerial qualities.
2. They can understand importance of engineer's role in management.
3. They will understand IT project management through life cycle of the project.
4. They will learn about project planning, execution, tracking, audit and closure of project.
5. They can understand processes in different departments of IT and non-IT industries
6. They can understand current technologies and future trends in IT Project Management.

UNIT – I   ENGINEERING MANAGEMENT   **6 Hours**
Engineering Management, Scientific Management, Administrative and Behavioral Management, Functions of technology management, Planning and forecasting, Decision making, Organizing, Leading Technical people, Controlling.

UNIT – II   PRODUCT MANAGEMENT   **6 Hours**
Managing Technology through product life cycle, Managing the research function, Product development process, Production planning and control systems, Product quality, Engineering ethics, Effectiveness achievement, Globalization, Managerial opportunities for Engineers.

UNIT - III   PROJECT PLANNING   **6 Hours**
Project initiation, proposals and contracts, Requirement specification, Project planning, task-set, scheduling and tracking of project, project resources, Estimation, Process definition and tailoring, Process Capability baseline, Effort estimation and scheduling, Quality planning and defect estimation, Risk Management.

UNIT - IV   PROJECT EXECUTION   **6 Hours**
Project Management plan, Project metrics, Configuration Management, Project execution, Life cycle execution, Peer review, Project monitoring and control, Time management, Cost management, HR Management, Communication management, Project Audits, Project closure, ISO 9000 to CMM, software process improvement.

UNIT – V   ERP BASICS   **6 Hours**
Introduction to ERP, Basic concepts of ERP, Risks and benefits of ERP, Functional modules, Market place dynamics, Related technologies, ERP implementation basics, Life cycle, Package selection, Strategies, Process of implementation, Consultants, vendors and employees, Success & Failure factors of ERP implementation, ERP operation and maintenance, Maximizing ERP system, ERP and E business.

UNIT - VI   PROJECT MANAGEMENT RECENT TRENDS   **6 Hours**

Professional Responsibility and personal excellence, Case studies to understand IT Project Management, Project Management tools, Future directions and trends in Areas like Software configuration Management, Supply Chain Management, Project Lifecycle Management, Business Process Management, Project Specific Documentation etc. Software Maintenance, Reengineering, Reverse Engineering

Text Books
1. Lucy, Morse, Daniel Babcock, "Managing Engineering and Technology", PHI Learning ISBN- 978-81-203-3825-8
2. Pankaj Jalote, "CMM in practice", Addison Wesley ISBN- 0-201-61626-2
3. Pressman, R. (2010). Software Engineering: A Practitioner's Approach, 7th or 8th Ed. Singapore: McGraw Hill. [ISBN -9780071267823]

Reference Books
1. Alexis Leon, "Enterprise Resource Planning", Tata Mcgraw-hill ISBN-978-0-07-065680-2
2. Jennifer Greene and Andrew Stellman, "Head First PMP", O'Relly, ISBN-978-0-596-10234-0
3. Tarek Khalil, "Management of Technology", Tata Mcgraw- Hill ISBN-978-0-07-067737-1
4. Shailesh Mehta, "Project Management and Tools and Technologies", SPD ,ISBN-9789351104520

314454: OPERATING SYSTEM LABORATORY

Teaching Scheme:
Practical : 4 Hours/Week

Examination Scheme:

Term-work : 50 Marks   Practical : 50 Marks

Prerequisites :
Data Structures

Course Objectives :
1.   To familiarize the students with the Operating System.
2.   To demonstrate the process, memory, file and directory management issues under the UNIX/LINUX operating system
3.   To introduce LINUX basic commands
4.   To make students how to make simple programs in LINUX and administrative task of LINUX

Course Outcomes :
1.   Describe OS support for processes and threads
2.   Recognize CPU Scheduling, synchronization, and deadlock.
3.   Use C / C++ and Unix commands, and develop various system programs under Linux to make use of OS concepts related to process synchronization, shared memory, file systems, etc.

Laboratory Assignments

1.   Process control system calls: The demonstration of fork, execve and wait system calls along with zombie and orphan states.
     a.   Implement the C program in which main program accepts the integers to be sorted. Main program uses the fork system call to create a new process called a child process. Parent process sorts the integers using merge sort and waits for child process using wait system call to sort the integers using quick sort. Also demonstrate zombie and orphan states.
     b.   Implement the C program in which main program accepts an integer array. Main program uses the fork system call to create a new process called a child process. Parent process sorts an integer array and passes the sorted array to child process through the command line arguments of execve system call. The child process uses execve system call to load new program that uses this sorted array for performing the binary search to search the particular item in the array.
2.   Thread management using pthread library.
     Implement matrix multiplication using multithreading. Application should have pthread_create, pthread_join, pthread_exit. In the program, every thread must return the value and must be collected in pthread_join in the main function. Final sum of row-column multiplication must be done by main thread (main function).
3.   Thread synchronization using counting semaphores and mutual exclusion using mutex.
     Application to demonstrate: producer-consumer problem with counting semaphores and mutex.
4.   Deadlock Avoidance Using Semaphores:
     Implement the deadlock-free solution to Dining Philosophers problem to illustrate the problem of deadlock and/or starvation that can occur when many synchronized threads are competing for limited resources.
5.   Inter process communication in Linux using following.

    a. **Pipes :** Full duplex communication between parent and child processes. Parent process writes a pathname of a file (the contents of the file are desired) on one pipe to be read by child process and child process writes the contents of the file on second pipe to be read by parent process and displays on standard output.

    b. **FIFOs:** Full duplex communication between two independent processes. First process accepts sentences and writes on one pipe to be read by second process and second process counts number of characters, number of words and number of lines in accepted sentences, writes this output in a text file and writes the contents of the file on second pipe to be read by first process and displays on standard output.

    c. **Signals :** Detecting the termination of multiple child processes :
Implement the C program to demonstrate the use of SIGCHLD signal. A parent process Creates multiple child process (minimum three child processes). Parent process should be Sleeping until it creates the number of child processes. Child processes send SIGCHLD signal to parent process to interrupt from the sleep and force the parent to call wait for the Collection of status of terminated child processes.

6. Linux Kernel configuration, compilation and rebooting from the newly compiled kernel.
   Requirements:
   a. Get a Linux kernel source code from www.kernel.org
   b. Menu based configuration of Linux kernel using menuconfig/xconfig/gconfig
   c. Creating a monolithic compressed image of a kernel
   d. Compilation of kernel modules
   e. Installation of kernel modules
   f. Finalize installation

7. Kernel space programming: Implement and add a loadable kernel module to Linux kernel, demonstrate using insmod, lsmod and rmmod commands. A sample kernel space program should print the "Hello World" while loading the kernel module and "Goodbye World" while unloading the kernel module.

8. Implement a new system call, add this new system call in the Linux kernel (any kernel source, any architecture and any Linux kernel distribution) and demonstrate the use of same.

9. Implementing a CPU scheduling policy in a Linux OS.
                                 OR
   Implementing a memory management policy in a Linux OS.
                                 OR
   Implementing a file system in a Linux OS.

All the assignments should be conducted on Latest version of Open Source Operating Systems, tools and   Multi-core CPU supporting Virtualization and Multi-Threading.

Links for Laboratory Assignments
http://homes.cs.washington.edu/~tom/nachos/
http://web.cecs.pdx.edu/~walpole/class/cse513/project/syscall.html
http://web.cecs.pdx.edu/~walpole/class/cse513/project/syscall.html

314455: SOFTWARE LABORATORY – II

Teaching Scheme:
Practicals : 4 Hours/Week

Examination Scheme:
Practical : 50 Marks   Oral : 50 Marks

Prerequisites : Discrete Structure, C/ C++ Programming

Course Objectives :
1. To learn the concepts of how to design and implement two pass assembler.
2. To learn use of macros and its expansion process.
3. To understand regular expressions and its applications in compiler design.
4. To learn the various algorithmic design paradigms.
5. To find the space and running time requirements of the algorithms.

Course Outcomes :
1. Be able to design and implement two pass assembler for hypothetical machine instructions.
2. Be able to design and implement the macroprocessor.
3. Be able to use the tools "Lex" and "YACC" for implementation of simple Calculator.
4. Will be able to apply algorithmic strategies for solving various problems such as travelling salesman problem, Hamiltonian circuit etc.

PART – I
1. Write a program to implement matrix multiplication using Strassen's method. (Divide and Conquer)
2. Write a program to implement coding and decoding using Huffman method. (Greedy) Students are expected to write the program with nlogn complexity and verify the same.
3. Write a program to print shortest path and cost for the directed graph using Floyd and Warshal method. (Dynamic Programming) and verify the complexity
4. Write a recursive program to find the solution of placing n queens on chess board so that no queen takes each other (backtracking).
5. Write a non-recursive program to check whether Hamiltonian path exists in undirected graph or not. If exists print it. (backtracking)
6. Write a program to solve the travelling salesman problem. Print the path and the cost. (Branch and Bound)

PART – II
1. Write a program to implement II pass assembler. (For hypothetical instruction set from Dhamdhere)
   a. Consider following cases only (Literal processing not expected)
   b. Forward references
   c. DS and DC statement
   d. START, EQU
   e. Error handling: symbol used but not defined, invalid instruction/register etc.
2. Write a program to implement II pass assembler. (For hypothetical instruction set from Dhamdhere). Consider Literal processing , forward references not expected
   a. Use of literals and not symbols
   b. LTORG, END
   c. Error handling

3. Write a program to implement Macro Processor. (AIF and AGO not expected, nested macro calls not expected)   with Error Handling: Macro Duplicate definition, Parameter Mismatching etc.
4. Write a program to convert RE to DFA. (Compiler point of view, RE to DFA direct method from Uho Ulman Sethi).
5. Write a program to implement calculator using LEX and YACC.
6. Intermediate code generation using LEX &YACC for Control Flow and Switch Case statements.

All the assignments should be conducted on Latest version of Open Source/Proprietary Operating Systems, tools and   Multi-core CPU supporting Virtualization and Multi-Threading.

314456 : SEMINAR& TECHNICAL COMMUNICATION LABORATORY

Teaching Scheme:
Practicals : 2 Hours/Week

Examination Scheme:
In-Semester Assessment
Termwork – 50 Marks

Introduction

Graduates of final year IT program are supposed to design and implement projects through knowledge and skills acquired in previous semesters. Students should identify complex engineering problems and find effective, efficient and innovative ways of solving them through their projects.

In a technical seminar, students should aim to review literature in a focused way for identifying a complex problem to be attempted in their final year project. Seminar should make the student attain skills like (a) gathering of literature in specific area in a focused manner (b) effectively summarizing the literature to find state-of-the-art in proposed area (c) identifying scope for future work (d) presenting (arguing) the case for the intended work to be done as project (e) reporting literature review and proposed work in scientific way using good English.

Prerequisites : Basic Communication and reading skills

Course Objectives :
1. To perform focused study of published or unpublish.ed technical/research literature relevant to a specific topic.
2. To study, interpret and summarize literature scientifically.
3. To build independent thinking on complex problems.
4. To build collaborative work practices
5. To communicate scientific information to a larger audience.
6. To use presentation standards and guidelines effectively.
7. To be ethical in presenting purpose, sources, conduct and use of the literature reviewed.
8. To define future work or problem statement for projects.

Course Outcomes :
1. Gather, organize, summarize and interpret technical literature with the purpose of formulating a project proposal.
2. Write a technical report summarizing state-of-the-art on an identified topic.
3. Present the study using graphics and multimedia techniques.
4. Define intended future work based on the technical review.

GUIDELINES FOR PROJECT BASED SEMINAR (PBS)
a. A group of 3-4 students shall identify problem(s) in Computer Engineering / Information Technology referring to recent trends and developments in consultation with institute guide.
b. The group must review sufficient literature (books, journal articles, conference papers, magazines, web resources etc.) in relevant area in parts (allocated to each student) decided by the guide.
c. Research articles could be referred from IEEE, ACM, Science direct, Springer, Elsevier, IETE,CSI or from freely available digital libraries like Digital Library of India (dli.ernet.in), National Science Digital Library, JRD Tata Memorial Library, citeseerx.ist.psu.edu, getcited.org, arizona.openrepository.com, Open J-Gate, Research Gate, worldwidescience.org etc.
d. The group shall present the study as individual seminars in 20 – 25 minutes concluding with the solution to problem identified as their project statement.

e. Each student shall submit two copies of the seminar report in a prescribed format duly signed by the guide and Head of the department.
f. Internal guide shall define a project statement based on the study by student group.
g. Broad contents of review report (20-25 pages) shall be
    i. Introduction of specific topic
    ii. Motivation, purpose and scope of review
    iii. Related work with citations
    iv. Discussion (your own reflections and analysis)
    v. Conclusions
    vi. Project definition. (Short version of RUP's vision document if possible).
    vii. Bibliography

EVALUATION SCHEME

a. A panel of staff members from the institute will assess the seminar internally during the presentation
b. Attendance for all seminars for all students is compulsory.
c. Criteria for evaluation
    i. Relevance of topic   - 05 Marks
    ii. Relevance + depth of literature reviewed - 10 Marks
    iii. Seminar report (Technical Content)   - 10 Marks
    iv. Seminar report (Language)   - 05 Marks
    v. Presentation Slides   - 05 Marks
    vi. Communication Skills   - 05 Marks
    vii. Question and Answers   - 10 Marks

Reference Books

1. Gersen and Gersen; Technical Writing: Process and Product; Pearson Education Asia.
2. Rutherfoord; Basic Communication Skills for Technology; Pearson Education Asia.
3. Lesikar et al; Lesikar's Basic Business Communication; Tata McGraw