SCHEME OF INSTRUCTIONS
BE III YEAR I SEMESTER
(COMPUTER SCIENCE AND ENGINEERING)

THEORY

Sl. No.	Syllabus Ref. No.	Subject
1.	C S 301	Database Management Systems
2.	C S 302	Operating Systems
3.	C S 303	Automata, Languages and Computation
4.	C S 304	Software Engineering
5.	C M 371	Managerial Economics and Accountancy
6.	C S 306	Data Communications

PRACTICALS

Sl. No.	Syllabus Ref. No.	Subject
1.	C S 331	Database Lab
2.	C S 332	OS Lab
3.	C S 333	Mini Project

CS 301

DATABASE MANAGEMENT SYSTEMS

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT – I

Introduction: Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Object–based and Semi-structured Databases, Data Storage and Querying, Transaction Management, Data Mining and Analysis, Database Architecture, Database Users and Administrators.

Database Design and the E-R Model: Overview of the Design Process, The Entity-Relationship Model, Constraints, Entity-Relationship Diagrams, Entity – Relationship Design Issues, Weak Entity Sets, Extended E-R Features, Database Design for Banking Enterprise, Reduction to Relational Schemas, Other Aspects of Database Design.

UNIT – II

Relational Model: Structure of Relational Databases, Fundamental Relational-Algebra Operations, Additional Relational – Algebra Operations, Extended Relational - Algebra Operations, Null Values, Modification of the Databases.

Structured Query Language: Data Definition, Basic Structure of SQL Queries, Set Operations, Aggregate Functions, Null Values, Nested Sub-queries, Complex Queries, Views, Modification of the Database, Joined Relations.

UNIT – III

Advanced SQL: SQL Data Types and Schemas, Integrity Constraints, Authorization, Embedded SQL, Dynamic SQL, Functions and Procedural Constructs, Recursive Queries, Advanced SQL Features. **Relational Database Design:** Features of Good Relational Design, Atomic Domains and First Normal Form, Functional-Dependency Theory, Decomposition using Functional Dependencies.

UNIT - IV

Indexing and Hashing: Basic Concepts, Ordered Indices, B+-tree Index Files, B-tree Index Files, Multiple-Key Access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices.

Index Definition in SQL Transactions: Transaction Concepts, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability.

UNIT – V

Concurrency Control: Lock-based Protocols, Timestamp-based Protocols, Validation-based Protocols, Multiple Granularity, Multi-version Schemes, Deadlock Handling, Insert and Delete Operations, Weak Levels of Consistency, Concurrency of Index Structures.

Recovery System: Failure Classification, Storage Structure, Recovery and Atomicity, Log-Based Recovery, Recovery with Concurrent Transactions, Buffer Management, Failure with Loss of Nonvolatile Storage, Advanced Recovery Techniques, Remote Backup Systems.

Suggested Reading:

1. Abraham Silberschatz, Henry F Korth, S Sudarshan, Database System Concepts, McGraw-Hill International Edition, 5th Edition, 2006
2. Ramakrishnan, Gehrke, Database Management Systems, McGraw-Hill International Edition, 3rd Edition, 2003
3. Elmasri, Navathe, Somayajulu, Fundamentals of Database Systems, Pearson Education, 4th Edition, 2004

CS 302

OPERATING SYSTEMS

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

Introduction to operating systems: OS structure and strategies, Process concept, Interprocess communication, Threads, Multithreaded programming.

Process scheduling: Scheduling criteria, Scheduling Algorithms, Multi Process scheduling, Thread Scheduling.

UNIT-II

Memory Management, swapping, contiguous allocation, paging, static and dynamic partition, demand paging, page replacement algorithms, thrashing, segmentation with paging, Virtual memory.

File System Interface: File Concept, Access Methods, Directory Structure, File System Mounting, File sharing, protection.

File System implementation: File system structure, File system implementations, Directory implementation, Allocation Methods, Free space management, Efficiency and performance, recovery.

Case Studies: UNIX file system, Windows file system

UNIT-III

Process Synchronization: Critical section problem, semaphore, monitors**.**

Deadlocks: Necessary conditions, resource allocation graph, methods for handling deadlocks, preventions, avoidance, detection and recovery, protection, goals of protection, domain of protection, access matrix .

UNIT-IV

Device Management: Disk structure, Disk Attachment, Disk Scheduling, Disk Management, RAID Structure, Stable storage implementation.
I/O System: I/O hardware, Application I/O interface, Kernel I/O Subsystem, Transforming I/O request to hardware operation, STREAMS.

UNIT-V

Case Studies:

LINUX System: Design Principles, Kernel Modules, Process Management, Scheduling Memory Management, File Systems, Input and Output, Interprocess communication, Network Structure, Security.

Windows XP: Design Principles, Architecture, Environmental subsystem, File Subsystem, Networking, Programming interface, Android OS

Suggested Reading:

1.Abraham Silberchatz, Peter B.Galvin, Greg Gagne, *Operating System-Concepts*, Wiley India, 2006.

2.Andrew S.Tanenbaum, *Modern Operating Systems*, Third Edition, Pearson education, Asia-2008.

3.DhananjayM.Dhamdhere, *Operating System-concept based approach*, third edition, Tata McGraw Hill, Asia-2009.

4. Robert Love: *Linux kernel Development*, Pearson Education, 2004.

CS 303

AUTOMATA LANGUAGES AND COMPUTATION

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

Automata: Introduction to Finite Automata, Central Concepts of Automata Theory. Finite Automata: An Informal Picture of Finite Automata, Deterministic Finite Automata, Non-deterministic Finite Automata, An application, Finite Automata with Epsilon Transitions.

Regular expressions & Languages: Regular Expressions, Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions.

UNIT-II

Properties of Regular Languages: Proving Languages not to be Regular, Closure properties of Regular Languages, Decision Properties of Regular Languages, Decision Properties of Regular Language, Equivalence and Minimization of Automata.

Context Free Grammars and Languages: Context free grammars, Parses Trees, Applications, Ambiguity in Grammars and Languages.

UNIT-III

Pushdown Automata: Definition, Languages of PDA, Equivalence of PDA's and CFG's Deterministic Pushdown Automata.

Properties of Context Free Languages: Normal Forms for Context Free Grammars, Pumping Lemma, closure properties, Decision Properties of CFL's.

UNIT-IV

Introduction to Turing Machines: Problems that Computers cannot Solve, The Turing machines, Programming Techniques for Turing Machines, Extensions to the Turing 4 Machines Restricted Turing Machines, Turing machines and Computers.

UNIT-V

Un-decidability: A language that is not Recursively Enumerable, An undecidable problem that is RE, Undecidable problems about Turing Machines, Post's Correspondence Problem,

Other Undecidable Problems. **Intactable Problems:** The Classes P and NP, an NP Complete Problem, A Restricted Satisfiability problem.

Suggested Reading :

1. 1.John. E. Hopcroft, Rajeev Motwani, Jeffery, D. Ulman, *Introduction*

to Automata Theory, Languages and Computation, 3nd edition, Pearson Education-2007.

1. 2.John C.Martin, *Introduction to Languages and the Theory of Computation,* 3rd edition Tata McGraw Hill, 2003.

1. 3.Bernard M.Moret, *The Theory of Computation,* Pearson Education,

CS 304

SOFTWARE ENGINEERING

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT -I

Introduction to Software Engineering:

Generic view of Process: Software Engineering, Process Framework, CMM, Process Patterns, Process Assessment, Personal and Team Process, Process Technology, Product and process.

Process Models: Perspective Models, Waterfall Model, Incremental Process Models, Evolutionary Process Models, Specialized Process Models, The Unified Process.

An Agile View of Process: What is Agility, Agile Process, and Agile Process Models.

UNIT-II

Planning and Managing the Project: Tracking Progress, Project Personnel, Effort Estimation, Risk Management, the Project Plan, Process Models and Project Management, Information Systems Example, Real-time Example.

Requirement Engineering: A bridge to design and construction, Requirement Engineering tasks, Initiating Requirement Engineering Process, Eliciting Requirement, Developing Uses cases, Building the Analysis Model, Negotiating Requirements, Validating Requirements.

UNIT-III

Building the Analysis Model: Requirements Analysis Modeling approaches, Data modeling concepts, Object oriented analysis , Scenario based modeling, Flow oriented modeling, Class-based modeling, Creating a Behavioral Modeling.
Design Engineering: Design with in the context of SE, Design Process and Design quality, Design concepts, The Design Model, Pattern-based Software Design.

UNIT-IV

Creating Architectural Design: Software architecture, Data design, Architectural Styles and Patterns, Architectural Design, Assessing alternative Architectural Designs, Mapping data flow into software Architecture.
Modeling Component-Level Design: What is a Component, Designing Class-Based components, Conducting Component–level Design, Object Constraint Language, Designing Conventional Components.

Performing User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Design Evaluation**.**

UNIT-V

Testing Strategies: A Strategic approach to software testing, strategic issues, test strategies for O-O software, validation testing, system testing, art of debugging.

Testing Tactics: Software Testing Fundamentals, Black-Box and white box Testing, basis path testing, Control Structure Testing, O-O Testing methods, Testing Methods applicable on the class level, inter class Test case design, Testing for Specialized environments, architectures and applications, Testing Patterns.

Product Metrics: Software quality, A framework for product metrics, Metrics for the analysis model, metrics for the Design model, metrics for source code, Metrics for Testing, Metrics for maintenance.

Suggested Reading:

1. 1.Roger S. Pressman, *"Software Engineering –A Practitioners Approach",* 6th Edition, Pearson Education, India, 2005.

1. 2.Shari Lawrence Pfleeger, *"Software Engineering Theory and Practices"* 4th Edition - Pearson Education, India, 2011.

1. 3.Pankaj Jalote, *"An Integrated Approach to Software Engineering",* 3rd Edition, Springer Link Edition, India, 2005.

CM 371

MANAGERIAL ECONOMICS AND ACCOUNTANCY

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

Meaning and Nature of Managerial Economics: Managerial Economics its usefulness to Engineers, Fundamental Concepts of Managerial Economics, Scarcity, Marginalism, Equi-marginalism, Opportunity costs, Discounting, Time Perspective, Risk and Uncertainty, Profits, Case study method.

UNIT-II

Consumer Behaviour: Law of Demand, Determinants, Kinds; Elasticity of Demand (Price, Income and Cross-Elasticity); Demand Forecasting, Law of Supply, Concept of Equilibrium. (Theory questions and small numerical problems can be asked).

UNIT-III

Theory of Production and Markets: Production Function, Law of Variable Proportion, ISO quants, Economics of Scale, Cost of Production (Types and their measurement), Concept of Opportunity Cost, Concept of Revenue, Cost-Output relationship, Break-Even Analysis, Price – Output determination under Perfect Competition and Monopoly (theory and problems can be asked).

UNIT-IV

Capital Management: Its significance, determination and estimation of fixed and working capital requirements, sources of capital, Introduction to capital budgeting, methods of payback and discounted cash flow methods with problems. (Theory questions are numerical problems on estimating working capital requirements and evaluation of capital budgeting opportunities can be asked).

UNIT-V

Book-keeping: Principles and significance of double entry book keeping, Journal, Subsidiary books, Ledger accounts Trial Balance, concept and
preparation of Final Accounts with simple adjustments, Analysis and interpretation of Financial Statements through Ratios.

(Theory questions and numerical problems on preparation of final accounts, cash book, petty cash book, bank reconciliation statement, calculation of some ratios).

Suggested Reading:

1. 1.Mehta P.L., "*Managerial Economics – Analysis, Problems and Cases"*, Sulthan Chand & Son's Educational publishers, 2011.

1. 2.Maheswari S.N. "*Introduction to Accountancy",* Vikas Publishing House, 2005.

1. 3.Panday I.M. "*Financial Management",* Vikas Publishing House,

CM 306

DATA COMMUNICATIONS

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

Introduction: Communication model, Data Communication networking, Protocols and Architecture, Standards.

Data Transmission: Concepts and terminology, Analog and Digital Transmission, Transmission Impairments, Transmission media.

Data Encoding: Digital Data Digital Signals, Digital Data-Analog Signals, Analog Data-Digital Signals, Analog Data-Analog Signals.

UNIT-II

Data Communication Interface: Asynchronous and Synchronous Transmission, Line Configuration, Interfacing.

Data Link Controls: Flow Control, Error Detection, Error Control, HDLC, other Data link Control protocols, performance issues.

UNIT-III

Multiplexing: Frequency Division Multiplexing, Synchronous time - Division Multiplexing, Statistical Time – Division Multiplexing. Asymmetric Digital Subscriber line, xDSL. Circuit Switching, Packet Switching & Frame Relay. ATM Architecture, Logical Connection, ATM Cells, Transmission of ATM cells.

UNIT-IV

Traditional Ethernet: Topologies and Transmission Media, LAN protocol architecture, MAC sub layer, - CSMA/CD, Physical Layer, Implementation, Bridged, switched and full duplex Ethernets, Layer 2 and Layer 3 Switches. Fast Ethernet: MAC sublayer, Physical sublayer, Implementation.

Gigabit Ethernet: MAC sublayer, Physical Layer, Implementation.

UNIT -V

Cellular Wireless Networks: Principles of Cellular Networks, First Generation Analog Second Generation CDMA, Third Generation Systems.

Wireless LANs: Overview, Wireless LAN Technology, IEEE 802.11. Architecture and services, IEEE 802.11 Medium Access Control, IEEE 802.11 Physical Layer.

Bluetooth: Architecture, Layers.

Suggested Reading:

1. 7.William Stallings, *Data and Computer communication,* 7th edition. Pearson Education, Asia-2004.

1. 8.Behrouz A. Forouzan, *Data Communications and Networking,* 4th Edition, Tata McGraw Hill, 2006.

1. 9.Fred Halsall, *Data Communications, Computer Networks and Open Systems,* 4th Edition, Pearson Education, 2000.

CS - 331

DATABASE MANAGEMENT SYSTEMS LAB

Instruction	3	Periods per week
Duration of University Examination	3	Hours
University Examination	50	Marks
Sessional	25	Marks

1. SQL

I. a.Creating Database (Exercising commands like DDL,DML,DCL and TCL)
II. b.Exercising all types of Joins
III. c.Creating tables in I Normal, II Normal, III Normal and BCNF Form.
IV. d.Creating table using combination of constraints.
V. e.Exercising Simple to Complex Queries
VI. f.Usage of Stored Functions.
VII. g.Creating Password and Security features for an Application.
VIII. h.Usage of File locking, Table locking facilities in an Applications

2. PL/SQL

a. a)Demonstration of Blocks, Cursors, Procedures, Functions and Packages.
b. b)Demonstrate Exception Handling .
c. c)Usage of Triggers to perform operation on Single and Multiple Tables.
d. d)PL/SQL Procedures for data validation

3. Report Generation Using SQL Reports

4. Creation of Small Full pledged Database Application

Note : *The creation of sample database for the purpose of the experiments is Expected to be pre-decided by the instructor.*

Suggested Reading :

1. 1.Nilesh Shah,
2. 2.Rick F Van der Lans,
3. 3.Benjamin Rosenzweig, Elena Silvestrova,
4. 4.Albert Lulushi,

CS 332

OPERATING SYSTEMS LAB

Instruction	3	Periods per week
Duration of University Examination	3	Hours
University Examination	50	Marks
Sessional	25	Marks

1.Printing file flags for specified descriptor.

2.Print type of file for each command line arguments

3.Recursively descend a directory hierarchy counting file types

4 .Program using process related system calls

5.Program s to create threads

6. Implement CPU scheduling algorithms (a) Round Robin (b) SJF

a. (c)FCFS

7.Implement page replacement algorithms (a) FiFo (b) LRU

8.Echo server using pipes

9.Echo server using messages

10.Producer- Consumer problem using shared memory.

11.Readers – Writers problem using message passing

12.Dinning philosopher problem using semaphore

13.Bankers algorithm for Deadlock detection and avoidance

14.Program using file locking

1. 15.Programs using LINUX shell scripts.

Case study of android OS

CS 333

MINI PROJECT

Instruction 3 Periods per week

Sessional 25 Marks

The students are required to carry out mini projects in any of the areas such as Data

Structures, Microprocessors and Interfacing, Database Management Systems, Operating Systems, Design and Analysis of Algorithms, and Software Engineering.

Students are required to submit a report on the mini project at the end of the semester.

SCHEME OF INSTRUCTIONS
BE III YEAR II SEMESTER
(COMPUTER SCIENCE AND ENGINEERING)

THEORY

Sl. No.	Syllabus Ref. No.	Subject
1.	C S 351	Web Programming & Services
2.	C S 352	Compiler Construction
3.	C S 356	Design & Analysis of Algorithms
4.	C S 354	Object Oriented System Development
5.	C S 355	Computer Networks

PRACTICALS

Sl. No.	Syllabus Ref. No.	Subject
1.	C S 381	Web Programming & Networking Lab
2.	C S 382	Object Oriented System Development Lab
3.	C S 383	Compiler Construction Lab
4.	C S 384	Mini Project

CS 351

WEB PROGRAMMING & SERVICES

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

Web basics and overview: introduction to Internet, World Wide Web, Web Browsers, URL, MIME, HTTP, Web programmer's toolbox Introduction to XHTML. Basics of Java script.

Introduction to XML, XML Document structure, DTD, namespaces, Schemas, XSLT style sheets, XML Processors

UNIT-II

The J2EE Platform: Enterprise Architecture styles, J2EE Architecture

– Containers, J2EE Technologies, Deploying J2EE applications Introduction to Web containers

Servlet Programming: Overview of Java Servlet API, Servlet implementation, Servlet Configuration, Servlet Exceptions, Servlet Life cycle, Request and Responses.

Servlet Sessions, Context, and collaboration: Approaches to Session tracking, Session Tracking with Java Servlet API, Servlet Context, Servlet Collaboration

UNIT-III

Filters for Web applications: What is a Filter, Sample Filter, Filter API, Deployment Descriptor for Filters, Chat Application with Filters.

Web Deployment, Authentication, and Packaging: Web application structure, Mapping requests to applications and servlets, Securing web applications, Deploying configuration.

JSP Basics and architecture: Introduction to JSP, Jsp Directives, Scripting Elements, Standard Objects, JSP Design strategies.
JSP Tag extensions: Tag extensions, A simple Tag Anatomy of a Tag extension, Writing Tag extensions, Application life cycle events

UNIT-IV

Java Mail: Mail protocols, Java Mail Overview, Installation and Configuration, Java mail API, working with Mail, Java mail resources

Database Programming with JDBC: Database Drivers

java.sql package: JDBC Process, Different types of statements, Retrieving meta information from Database and ResultSet

javax.sql package: JDBC Data sources, Connection pooling, Distributed transactions, RowSet objects

UNIT-V

.NET Platform: Introduction to .NET Framework, Common type systems, Common Language Runtime.

Introduction to C#, Types and Objects, Program structure.

Introduction to ASP.NET: The basics, ASP.NET documents, Code behind files

ASP.NET controls- HTML controls, Life cycle, page level events, control events, web controls, creating controls with in code, Response output for controls, validation controls

.NET Remoting, Database Connectivity with ADO.NET

Suggested Reading:

1. 1.Subramnyam Allamraju, *Professional java server programming J2EE* 1.3 Edition, Cedit Buest. Apress Publications

1. 2.Robert W Sebesta, *Programming the World Wide Web,* Pearson Education

1. 3.Joe Duffy, *Professional .NET Framework 2.0,* Wiley India

CS 352

COMPILER CONSTRUCTION

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

Introduction – Programs related to compilers. Translation process. Major data structures. Other issues in compiler structure. Boot strapping and porting.

Lexical analysis – The role of Lexical Analyzer. Input Buffering. Specification of Tokens. Recognition of Tokens. The Lexical-Analyzer Generator Lex.

UNIT-II

Syntax Analysis – Introduction. Top-Down parsing, Brute Forcing, Recursive Descent, Predicative LL(1), Bottom-Up parsing : Introduction to LR Parsing, Powerful LR parsers SLR, CALR, LALR, Using Ambiguous Grammars, Parser Generators - Yacc.

UNIT-III

Syntax Directed Translation – Syntax Directed Definitions. Evaluation Orders for SDDs. Applications of Syntax Directed Translation.

Symbol Table Organization - Structure of Symbol table, Symbol Table organization for Block Structured and non block Structure languages, Data Structures of symbol Table.

UNIT-IV

Intermediate code generation : Variants of syntax trees. Three-Address Code, Types and Declarations. Translation of Expressions. Type Checking. Control Flow.

Storage Organization. Stack Allocation of Space. Access to Non local Data on the Stack. Heap Management. Introduction to Garbage Collection.

UNIT-V

Code Generation – Issues in the Design of a Code Generator. The Target Language. Addresses in the Target Code Basic Blocks and Flow Graphs. Optimization of Basic Blocks. Peephole Optimization. Register Allocation and Assignment. Machine Independent Optimizations – The Principal Sources of Optimizations, Introduction to data flow analysis, Foundation of data flow analysis.

Error Recovery : Introduction, Error detecting and Reporting in various Phases, Lexical Errors, Syntax Errors handling, and error Recovery in various Phases.

Suggested Reading:

1. 1.Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D Ullman –

Compilers: Principles, Techniques & Tools, Pearson Education 2nd Edition 2007.

1. 2.Keith D Cooper & Linda Tarezon, *Engineering a Compiler,* Morgan Kafman, Second edition.

1. 3.Lex & Yacc, John R Levine, Tony Mason, Doug Brown, Shroff Publishers.

1. 4.Kenneth C Louden, *Compiler Construction: Principles and Practice*, Cengage Learning.

Lex & Yacc, John R Levine, Oreilly Publishers.

CS 356

DESIGN AND ANALYSIS OF ALGORITHMS

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

Introduction, Algorithm Specification, Performance analysis, Space Complexity, Time Complexity, Asymptotic Notation(O,Omega,Theta), Practical Complexities, Performance Measurement, Review of elementary data structure- Heap and Heap Sort, Hashing, Set representation. UNION, FIND.

UNIT-II

Divide-and Conquer: The general method, finding maximum minimum. Merge sort quick sort and selection.

Greedy Method: Knapsack problem, Optimal Storage on tapes, Job sequencing with deadlines, Optimal merge patterns, Minimum Spanning Trees.

UNIT-III

Dynamic Programming And Traversal Technique: Multistage graph, All Pair Shortest Path, Optimal Binary Search trees,0/1 Knapsack, Reliability Design, Traveling Salesman Problem, Bi connected Components and Depth First Search.

UNIT-IV

Backtracking and Branch and Bounds: 8-Queens Problem, Graph Coloring Hamilton cycle, Knapsack Problem, 0/1 Knapsack Problem, Traveling salesperson problem, Lower-Bound Theory.

UNIT-V

NP-Hard and NP-Completeness: Basic concepts, cook's theorem, NP-hard graph problems and scheduling problem, NP-hard code generation

Logic Programming Concepts, Prolog, Theoretical Foundations, Logic Programming in

Perspective

problems, Clique Decision problem, Node covering decision, Scheduling problem, NP hard code generation problem.

Suggested Reading:

1.Horowitz E. Sahani S: *"Fundamentals of Computer Algorithm",*

Galgotia Publications.

2.Anany Levitin, *"Introduction to the Design & Analysis, of Algorithms",* Pearson Education, 2000.

3. Aho, Hopcroft, Ulman, *"The Design and Analysis of Computer Algorithm",* Pearson Education, 2000.

1. Parag H. Dave, Himanshu B. Dave *"Design and Analysis of Algorithms"* Pearson Education, 2008.

CS 354
OBJECT ORIENTED SYSTEM DEVELOPMENT

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

UNIT-I

UML Introduction : Necessity of a Model, Introducing the UML, Hello World.

Basic Structural Modeling: Classes, Relationships, Common Mechanisms,

Diagrams, Class Diagrams.

Advanced Structural Modeling: Advanced Classes, Advanced Relationships,

Interfaces, Types and Roles, Packages, Instances, Object Diagrams, Components.

UNIT-II

Basic Behavioral Modeling: Interactions, Use Cases, Use Case Diagrams, Interaction diagrams, Activity diagrams.

Advanced Behavioral Modeling: Events and Signals, State Machines, Processes and Threads, Time and space, State Chart Diagrams.

UNIT-III

Architectural Modeling: Artifacts, Deployment Collaborations, Patterns and Frame-works, Artifact diagrams, Deployment diagrams, Systems and models.

UNIT-IV

Unified Software Development Process: The Unified Process, The Four Ps, A

Use-Case-Driven Process, An Architecture-Centric Processes, An Iterative and

Incremental Process.

UNIT-V

Core Workflows: Requirements Capture, Capturing Requirements as Use Case,

Analysis, Design, Implementation, Test.

Suggested Reading:

1. 1.Grady Booch, James Rumbaugh, Ivor Jacobson, *"The Unified Modeling Language-User Guide"* (Covering UML 2.0), 2nd Edition, Pearson Education, India,2007.

1. 2.Ivor Jacobson, Grady Booch, James Rumbaugh: *"The Unified Software Development, Process"* Pearson Education, India, 2008.

CS 355

COMPUTER NETWORKS

Instruction	4	Periods per week
Duration of University Examination	3	Hours
University Examination	75	Marks
Sessional	25	Marks

Objectives:

- To understand the state-of-the-art technology in network protocols, network architecture and networked systems
- To learn the design principles of network infrastructure
- To gain proficiency in network programming

Outcomes:

This course enables the student to develop and demonstrate the following

- Ability to compare different network architectures
- Ability to understand the design principles of networking
- Ability to develop applications using network programming

UNIT-I

Review of ISO OSI Reference Model and TCP/IP Architectures.
Network Layer: Design issues, Services, Internal organization, Comparison of Virtual circuits and Datagram subnets. Routing Algorithms: The Optimality principle, Shortest path routing, Flooding, Flow-based algorithms, Distance vector, Link state, Hierarchical algorithms, Broadcast and Multicast routings. Congestion control algorithms: General principles, Traffic shaping, Congestion control in virtual circuit subnets, Choke packets, Load shedding, Jitter control and Congestion control for multicasting, Quality of Service (QoS)

UNIT-II

Internet working: How networks differ, Concatenated virtual circuits, Connectionless internet working, Tunneling, Internetwork routing, Fragmentation and Firewalls.
The Network Layer of the Internet: The IP protocol, IP addresses, Subnets, Internet control protocols, Gateway routing protocols, Multicasting, CIDR.

UNIT-III

Transport Layer: Service primitives, Addressing, Establishing a connection, Releasing a connection, Flow control, Buffering, Multiplexing and Crash recovery.
Internet Transport Protocols (TCP and UDP): The TCP service model, The TCP protocol, The TCP Segment Header, TCP connection management, Transmission policy: Congestion control, Timer management and UDP, Performance issues.

UNIT-IV

Application Layer:
Domain Name System: DNS name space, Resource records, Name services.
SMTP and MIME, HTTP, SNMP, Telnet, ftp, Multimedia.

UNIT-V

Socket programming: Socket address, Elementary socket system calls, Advanced socket system calls, Reserved ports, Socket options, Asynchronous I/O, Input/Output Multiplexing, Out-of-Band data, Sockets and Signals, Internet Super Server, DNS.

Suggested Reading:

1. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, 5th Edition, Pearson, 2012
2. Chwan-Hwa (John) Wu, J. David Irwin, Introduction to Computer Networks and Cyber Security, CRC Press, 2013

3. James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach Featuring the Internet, 5th Edition, Addison-Wesley, 2012
4. W. Richard Stevens, Unix Network Programming, Prentice Hall/Pearson Education, 2009
5. W. Richard Stevens, Andrew M Rudoff, Bill Fenner, Unix Network Programming: Networking APIs: Sockets and XTI (Volume 1) 3rd Edition, PHI

CS 381

WEB PROGRAMMING AND NETWORKING LAB

Instruction	3	Periods per week
Duration of University Examination	3	Hours
University Examination	50	Marks
Sessional	25	Marks

Web Programming Experiments:

1. 1.Creation of static web site using XHTML.
1. 2.Demonstration of XML, XSLT.
1. 3.Validation of static web page using Java script.
1. 4.Creation of dynamic content in web application using servlets.
1. 5.Handling Sessions in web applications.
1. 6.Usage of Filters in web applications.
1. 7.Creation of dynamic content in web application using JSP.
1. 8.Creation of dynamic content in web application using ASP.NET
2. 9.Providing data store support for web site using JDBC

Network Programming Experiments:

1. 10.Understanding and using the following commands. Ifconfig, netstat, ping, arp, telnet, tftp, ftp.
1. 11.Implementation of concurrent and iterative Echo server using both connection oriented and connectionless Socket System Calls.
2. 12.Implementation of time of the day service as Connection Oriented Concurrent Server using Socket System Calls.
1. 13.Build a concurrent Multithreaded File Transfer Server. Use separate Threads to allow the server to handle multiple clients concurrently.
2. 14.Implementation of Remote Program execution using Socket system calls. Programs to demonstrate the usage of Advanced Socket System calls Like Getsockopt(), Setsockopt(), Select(), Readv(), getpeernanel(), Getsockname()
1. 15.Implement a Concurrent Chat Server that allows currently logged in users to communicate with one another. Use Socket System

CS 382

OOSD LAB

Instruction	3	Periods per week
Duration of University Examination	3	Hours
University Examination	50	Marks
Sessional	25	Marks

Select one large Information System/Approach and device the following using CASE TOOL.

1. Data Flow diagram. 2. E-R diagram.
1. 3.Dynamic Model and Using Finite State Automata.
1. 4.Software Requirement Specification Document (SRS)
1. 5.Functional Decomposition and Structure.

6.	Data Dictionary.	
8.	Test Data Generation.	

1. 10.Verification
1. 11.User Manual
1. 12.Study of Software Maintenance Tools (SCCS, Debug Tools).
1. 13.A case study using Case Tool supporting

CS 383

COMPILER CONSTRUCTION LAB

Instruction	3	Periods per week
Duration of University Examination	3	Hours
University Examination	50	Marks
Sessional	25	Marks

1.Scanner programs using C

2.Scanner programs using LEX

3.Finding first set and follow set of productions

4.Top down parsers (Recursive decent parser, LL(1) parser, etc.)

5.Bottom up parsers (LR, SLR etc.)

6.Parser programs using YACC

7.Intermediate code generation

8.Code

CS 384

MINI PROJECT

Instruction	3	Periods per week
Sessional	25	Marks

The students are required to carry out mini projects in any of the areas such as Data Communications, Web Programming & Services, Computer Networks, Compiler Construction, and Object Oriented System Development.

Students are required to submit a report on the mini project at the end of the semester.