Parvatibai Chowgule College of Arts and Science
(Autonomous)
Margao, Goa

M.Sc.[I.T.]

Course Structure of MSc IT

SEMESTER I (24 CREDITS)

Course Code	**Course Type**	**Course Name**	**Credits**	**Student (hrs/week)**		
				L	**T**	**P**
MIT 11	Core – I	Data Structures and Algorithms	4	4	0	0
MIT 12	Core – II	Applied Probability and Statistics	4	4	0	0
MIT 13	Core – III	Operating Systems and Networks	4	4	0	0
MIT 14	Elective – I	Elective Course	4	4	0	0
MIT 15	Soft Skills – I	Communication Skills Course	4	4	0	0
MIT 16	Lab – I	Data Structures and Algorithms Lab	2	0	0	6
MIT 17	Lab – II	Operating Systems and Networks Lab	2	0	0	6

SEMESTER II (22 CREDITS)

Course Code	**Course Type**	**Course Name**	**Credits**	**Student (hrs/week)**		
				L	**T**	**P**
MIT 21	Core – IV	Software Architecture, Design Patterns and Frameworks	4	4	0	0
MIT 22	Core – V	Design and Analysis of Algorithms	4	4	0	0
MIT 23	Core – VI	Advanced Database Management Systems	4	4	0	0
MIT 24	Elective – II	Elective Course	4	4	0	0
MIT 25	Lab – III	Software Architecture, Design Patterns and Frameworks Lab	2	0	0	6
MIT 26	Lab – IV	Design and Analysis of Algorithms Lab	2	0	0	6
MIT27	Lab-V	Advanced Database Management Systems Lab	2	0	0	6

SEMESTER III (22 CREDITS)

Course Code	**Course Type**	**Course Name**	**Credits**	**Student (hrs/week)**		
				L	**T**	**P**
MIT 31	Core – VII	Data Mining	4	4	0	0
MIT 32	Core – VIII	Information Retrieval	4	4	0	0
MIT 33	Elective – III	Elective Course	4	4	0	0
MIT 34	Elective – IV	Elective Course	4	4	0	0
MIT 35	Elective – V	Elective Course	4	4	0	0
MIT 36	Lab VI	Data Mining & Information Retrieval Lab	2	0	0	6

SEMESTER IV(12 CREDITS)

Course Code	**Course Type**	**Course Name**	**Credits**	**Student (hrs/week)**		
				L	**T**	**P**
MIT 41		Dissertation	12	0	0	0

Legend:

L– Lectures
T – Tutorials
P – Practicals

Detailed Syllabus

Course Title: Data Structures and Algorithms

Course Code: MIT 11

Marks: 100

Credits: 4

Course Pre-Requisites:

An introductory course on Data Structures

Course Objectives:

- The objective of the course is to understand the real-life applications of data structures.
- Be familiar with writing recursive and iterative methods using Data Structure.

Learning Outcome:

- Have an idea of applications of algorithms in a variety of areas such as game theory etc.
- To make foundation of writing programs using algorithms on trees,graphs etc.

Syllabus:

1: Algorithm Analysis: [5L] Mathematical Background, Big-O notation, RunningTime computation, Introduction to different strategies of algorithm design – Divide and Conquer, Greedy, DynamicProgramming etc.

2:Review of Basics Data Structure: **[4L]**

List, Doubly linked list, Circular list, Stack, Queue, Recursion.

3:Trees: **[17L]**

The Huffman algorithm, Representing list and binary trees, Height balanced tree, Dictionaries, Optimal Binary Search Trees,AVL Tree, Red Black Tree, B tree, B+ tree, suffix trees,Splay trees,Binary Tries, Compressed Binary Tries. Tries and Packet Forwarding,Quad Trees,R-Trees.

4:Sorting and searching: **[10L]**

Radix Sort,Heap sort, Quick Sort, Merge Sort ,Interpolation search,Tree Searching,General Search Trees,Hashing,External Sorting Algorithm.

5:Graphs and their applications: [12L]

Revision of basic graph traversal and search technique,spanning trees,Cut-sets and Cut-Vertices: Cut-Sets, Properties of Cut-sets, All Cut-sets in a Graph ,Eulerian Graphs , Hamiltonian Graphs

6:Storage management: [6L]

General list, Automatic list management, Dynamic memory management.

7:Problem classification: [3L]

Nondeterministic algorithm, The class of P, NP, NP-hard and NP- Complete problems

8: Introduction to Approximation and Randomized Algorithms [3L]

Absolute Approximation,€ approximations and randomization

List of Books:

1. Aaron M. Tenenbaum & Augenstein "Data structures using C and C++" PHI
2. Sartaj Sahani "Data structures, Algorithms and Applications in C++"
3. Alfred V. Aho, John E. &: Hoproft, Jeffrey D. Ullman, "Data structures and algorithms"
4. Advanced Data Structures by Peter Brass
5. Graph Theory with Applications to Engineering and Computer Science, Narasingh Deo, PHI.

Course Title: Applied Probability and Statistics

Course Code: MIT 12

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- To provide the foundation of Probability theory and Statistical inference in order to apply statistical methods to various fields such as Statistical Quality Control.

Learning Outcomes:

- Gain knowledge about the probability theory and statistical inference.
- Provide an outline of statistical quality control.

Syllabus:

1. **Introduction: [6L]** Probability models, sample space events, algebra of events, graphical methods of representing events, probability axioms, combinational problems, conditional probability, independence of events, Baye's rule, Bernoulli trials

2. **Discrete Random Variables: [10L]**
Introduction, random variables and their spaces, the probability mass function, distribution functions, special discrete distributions, analysis of program, the probability generating function, Discrete Random Vectors, independent random variables

3. **Continuous Random Variables: [10L]**
Introductions, the exponential distribution, some important distribution, functions of a random variable, jointly distributed random variables, distributions of sums, functions of normal random variables

4. **Expectation: [10L]** Introduction moments, expectation of functions of more than one random variable, moments and transforms of some important distributions, computations of mean time to failure, inequalities and limits theorems

5. **Conditional Distribution and Conditional Expectation: [2L]**
Conditional Expectation

6. **Statistical Inference: [14L]**
Introduction, Parameter Estimation, Hypothesis testing: z, t, chi square, F test, Regression, correlation and 'analysis of variance: Introduction, least squares curve fitting, the coefficient of Determination, confidence Intervals in linear Regression, correlation analysis, simple nonlinear regression, Higher dimensional least-squares fit, Analysis of variance; Non parametric tests: sign test, u test, Rank test, Median test

7. **Statistical Quality Control: [8L]**
Control charts, Mean chart, R chart, sigma chart, C chart

List of Books:

1. Gupta S.G.,and V.K.Kapoor, Introduction to Probability and Statistics
2. Sheldon M. Ross, Probability
3. P. S. Mann, Introduction to Statistics, Willey Student Edition

Course Title: Operating Systems and Networks

Course Code: MIT 13

Marks: 100

Credits: 4

Course Pre-Requisites:

- Basics of Operating Systems and Networks

Course Objectives:

- To understand Real time operating systems
- To gain understanding in specific areas of networking such as the design and maintenance of individual networks.

Learning Outcomes:

- Understand real time operating systems.
- Understand and building the skills of subnetting and routing mechanisms. Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.

Syllabus:

1. Overview of Operating Systems [5L]

Processes and Threads - Process Scheduling -Synchronization Mechanisms –Deadlocks: Detection, Prevention and Recovery – Models of Resources – MemoryManagement Techniques.

2. Real time Operating systems [10L]

Basic model of real time systems, Characteristics, Applications of real time systems, Real time task scheduling, handling resource sharing, Mobile operating systems, Micro kernel design, Processes and Threads, Memory Management, File system.Failure Recovery and Fault Tolerance:Types of faults, Issues, Failed system behavior, Failure detection, Approaches of fault tolerance

3. Overview of Network Service Design [4L]

Introduction, Strategy for Network Service Implementation, Issues in Network design

4. TCP/IP [8L]

Introduction to TCP/IP, Benefits of using TCP/IP,IP addressing, IP Network and Host addressing,Classfull and classless IP addresses, IPV6, Subnet mask, Subnet ting and super netting

5. Switch Technology [5L]

Switch fundamentals (Bridges vs. Switches) **–** Spanning Tree Protocol: Overview, Spanning tree protocol, Rapid Spanning tree protocol.

6. VLANs and VLAN Trunking [8L]

VLAN- concepts, broadcast domains with VLANs and routers, benefits, types, configuration (geographic, static, verifying, saving, deleting, troubleshooting), preventing broadcast storms. VLANTrunking Protocol, VTP modes of operation, Routing between VLANs, Inter-VLAN routing-issues, solutions, interfaces (physical, logical), subinterfaces.

7. Routing [8L]

Static V/s Dynamic routes, Adding and deleting static routes

Demand on dial routing, Routing protocol, RIP, OSPF, IGP, Secure IP routing.

8. **Network administration** **[8L]**

SNMP & RMON - Overview and features, MIB Management Information base

Installing SNMP Servers, SNMP communities, Authentication and securing

Monitoring and analysis and troubleshooting, Overview and installation configuration: fire wall, NAT, E-mail (Send mail), Radius, Remote access servers, proxy servers.

9. **Wireless Networking** **[**4L]

Overview, Infrastructure mode, Ad Hoc Mode, ESSID, wireless channels, wireless security, Authentication.

List of books:

1. MukeshSinghal and N.G Shivaratri, “Advanced concepts in operating systems”, McGraw-Hill 2000.
2. Jeffrey S. Beasley and PiyasatNilkaew, A practical guide to advanced networking,Pearson.
3. William Stallings, "Wireless Communications and networks" Pearson / Prentice Hallof India, 2nd Ed., 2007.

Course Title: Communication Skills Course

Course Code: MIT 15

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- To understand the essential elements of Written Communication, and the process of writing.
- To learn various subgenres of workplace communication, including business & technical writing
- To learn the dynamics involved in oral communication, including non-verbal interaction
- To use language effectively in public oral communication

Learning Outcomes:

- To apply creative thinking abilities necessary for effective communication in the modern workplace situation
- To demonstrate clarity, precision, conciseness and coherence in use of language
- To learn how to make one's writing better, faster and more successful
- To produce successful documents in any given situation in different formats, while considering the writer’s objectives, the reader’s needs, the reader-writer relationship and the context.
- To increase personal confidence in delivering speeches to small & large audiences
- To understand and gain non-verbal skills essential to effective speaking.
- Make proper presentations that disseminate information, conduct

negotiation and use persuasion

1. Introduction & theory of Written Communication [6L]

Process of Communication, Language as a Tool of Communication, Levels of Communication, Flow of Communication, Communication Networks, Barriers to Communication

2. The Writing process [5L]

Features of academic communication, Prewriting (Invention), Stasis Theory, Creating a Thesis Statement, Developing an Outline, Proofreading, Avoiding Plagiarism

3. Constituents of Effective Writing [8L]

Words & phrases, Sentence Construction, Paragraph Development, Précis Writing, Reading Comprehension

4. Business Writing [9L]

Letters, Memos, Emails, Proposals, Reports, Analysis and Presentation of Data, Documentation and Document Design

5.Technical Writing [4L]

Defining Technical Writing, Technical Description, Process Description, Instruction Manuals, User Manuals, Audience Awareness

6. Introduction to Oral Communication [6L]

Theory of Verbal Communication, Features of Verbal Communication, Listening Skills

7. Non-verbal communication [6L]

Kinesics, Proxemics, Paralinguistics, Chronemics

8. Public Speaking [8L]

Preparation for Public Speaking, Speech Writing, Delivery of Speech, Anxiety Management

9. Meetings / group activity [8L]

Interviews, Group Communication and Discussion, Team work, Leadership Skills

Course Title: Data Structures and Algorithms Lab

Course Code:MIT 16

Marks:50

Credits:2

Course Pre-Requisites:

- Theoretical Knowledge of Data Structures

Course Objective:

- Introduce students to a number of highly efficient algorithms and data structures for fundamental computational problems across a variety of areas.

Learning Outcome:

- Knowledge of key issues in advanced data structures, such as appropriateness of data structures and efficiency of their related algorithms.
- Proficiency in applying knowledge from the theory of advanced data structures to various application areas.

List of suggested assignments:

1.Implementation of Basic Data Structure such as stack,Queue,Linked List etc. [3P]

2. Write a program to implement self organised linked list. [2P]

3. Write a program to implement different sorting techniques. [2P]

4. Write a program to implement Optimal Binary Search Tree. [2P]

5. Write a program to generate a Huffman code for the text file. [2P]

7. Write a program to implement insertion and deletion in AVL Tree. [2P]

8. Write a program to implement Red Black tree. [2P]

9.Write a program to implement B-Tree. [2P]

10. Write a program to implement B+ Tree. [2P]

11. Implementation of R-Tree and Quad Tree. [3P]

12. Implementation of Tries. [2P]

11.Implementation of Compressed Binary Tries. [2P]

11. Implementation of Dictionaries. [2P]

13. Implementation of Graph Traversal Techniques. [1P]

14. Write a program to implement linear probing, quadratic hashing and Double hashing. [1P]

Mini Project on the application of Data Structures.

Course Title: Operating Systems and Networks Lab

Course Code: MIT 17

Marks: 50

Credits: 2

Course Pre-Requisites:

- Theoretical Knowledge of operating systems and networks.

Course Objectives:

- To provide practical base in operating system and networks.

Learning Outcomes:

- Understand the basic structure and functioning of operating system.
- Implement various networking concepts**.**

Course contents:

1. Write a program to implement File I/O and efficiency, file system calls, dup functions, fcntl, stat functions, set uids, permissions, sticky bit, links, file times and utime function, directory creation and reading, mkdir, chdiretc, special files. [3P]

2. Implement Streams and file objects, positioning, reading and writing to streams, binary I/O, formatted I/O. [3P]

3. Implement Identifiers, fork, exec, wait functions, race conditions, changing user and group ids, process accounting and times, controlling terminals. [3P]

4. Implement an iterative TCP client and server application (eg. transfer file). [2P]

5. Implement a concurrent TCP client and server application to transfer file. [2P]

6. Implement UDP client and server application to reverse the given input sentence. [2P]

7. Using port scanning to identify open ports and vulnerability detection. Using nmap command. [2P]

8. Installing and configuring a Firewall. [2P]

9. Installing and configuring Intrusion detection system. [2P]

10. Using Network protocol analyzer tool like ethereal or tcpdump to analyze network traffic. [3P]

11. Creating subnets and supernets using simulation tools. [2P]

12. Configuring static route and dynamic route using routing tools. [2P]

13. Configuring VLANs. [2P]

Course Title: Software Architecture, Design Patterns and Frameworks

Course Code: MIT 21

Marks: 100

Credits: 4

Course Pre-Requisites:

- Introductory course on Software Engineering

Course Objectives:

- Learning Software Development using good OO Design and Architecture
- Understanding of Design and Architectural patterns and Frameworks.

Learning Outcomes:

- Understand various design patterns and their application in Software Development
- Understand software architecture and frameworks
- Understand antipatterns and steps that should not be taken while developing software

Syllabus:

1. **Principles of good OO design** **[6L]**
   Inheritance versus delegation, program to an interface.

2. **Design patterns** **[8L]**
   Motivation, reusability, extendibility, cataloging patterns, “GoF” patterns

3. **Software architecture** **[8L]**
   Definition, advantages, components and connectors, views, documenting, evaluating, mining.

4. **Architectural patterns:** **[8L]**
   Layered, pipe & filter, MVC, broker, microkernel, broker, peer to peer, plug-in, event bases software, SOA, middleware architectures

5. **Frameworks:** **[10L]**
   Enterprise frameworks, EJBs

6. **Software product lines:** **[6L]**
   Economies of scope, product line development, product development

7. **Model driven architecture:** **[6L]**
   PIM, PSM, transformation, software factories:

8. **Anti-patterns** **[8L]**
   Case studies

LIST OF BOOKS:

1. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Pearson Education

2. Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice, Pearson Education

3. Frank Buchmann, Regine Munier, Hans Rohnert, Peter Sommerland, Michael Stahl, Pattern Oriented Software Architecture-I, Pearson Education

Course Title: Design and Analysis of Algorithms

Course Code: MIT 22

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- Understand the basic concepts related to the design and analysis of algorithms
- Understand classical algorithms and their complexity
- Apply the algorithms to real-world problems

Learning Outcomes:

- To explain basic concepts related to the design and analysis of algorithms

- To describe classical algorithms and their complexity.
- To design and analyse their own algorithms.

Syllabus:

1: Introduction
[9L]

The Role of Algorithm in computing, Framework for design and analysis of algorithms, Growth of functions: asymptotic notation; Recurrences: substitution method, recursion-tree method, master method; Probabilistic analysis and randomized algorithms, indicator random variables. Advanced Design and Analysis Techniques

2:Dynamic programming:
[9L]

Assembly line scheduling, matrix-chain multiplication, elements of DP, longest common subsequence,Optimal BST.

3: Greedy
algorithms: [7L]

elements of greedy strategy, Huffman codes,.Optimal storage on tapes,Minimum cost spanning tree- Kruskal and Prim's algorithms, performance analysis.

4: Backtracking [8L]

The general method,8 Queens problem , sum of subsets,Graph coloring,

5: Amortized analysis:
[3L]

Aggregate analysis, accounting method, potential method, dynamic tables

6: Graph Algorithms :
[8L]

Elementary graph algorithms- Minimum spanning tree: growing a spanning tree, Single-source shortest paths: Bellman-ford algorithm, Dijkstra's algorithm. All pairs shortest paths: shortest paths and matrix multiplication, floyd-warshall algorithm.

7: Internet Algorithms [7L]

Strings and patterns matching algorithm. Tries.
Text compression.Text similarity testing.

8: NP-Completeness: [4L]

Polynomial time, polynomial time verification, NP-completeness and reducibility

9: Approximation algorithms: [5L]

The vertex cover problem, Traveling salesman problem, the set-covering problem

LIST OF BOOKS:

1. Cormen Thomas, L. Charles, R. Ronald, S. Clifford, "Introduction to Algorithms", Second Edition, EEE, PHI
2. Knuth Donald, "The Art of Computer Programming Vol I, II, III", Addison Wesley
3. Ellis Horowitz, Sartaj Sahni & Sanguthevar Rajasekaran, Computer Algorithms, Galgotia, 2nd Edition
4. A. Aho, J. Hopcroft, & J. Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley
5. Richard Gilberg & Behrouz Forouzan, Data Structure: a Pseudo code Approach

Course Title: Advanced Database Management Systems

Course Code: MIT 23

Marks: 100

Credits: 4

Course Pre-Requisites:

- Introductory course on Database Management Systems

Course Objectives:

- Understand the concept of a database transaction and related database facilities, including concurrency control, journaling, backup and recovery, and data object locking and protocols.

Learning Outcomes:

- Critically evaluate alternative designs and architectures for databases and data warehouses

- Discuss and evaluate methods of storing, managing and interrogating complex data

- Analyze the background processes involved in queries and transactions, and explain how these impact on database operation and design

Syllabus:

1: Design Theory for Relational Database:[7L]

Functional Dependencies, Decomposition of relational schemes, Normal forms for Relations, schemas, Multivalued and other forms of Dependencies.

2: Query Processing and Optimization: [9L]

Basic algorithms for executing query operations, Basicoptimization strategies, Algebraicmanipulations, optimization of selections in system, Exact optimization for a subset of relational queries, Optimization under weak equivalence.

3: Concurrent operation on the database: [9L]

Basic concepts, a simple transaction model, serializability, lock based protocols, Timestamp based protocol, concurrency for hierarchically structured items, Deadlock handling (Wait-die, wound-wait, nowaiting, cautious waiting), optimistic concurrency control.

4: Database Recovery Techniques:[8L]

NO-UNDO/REDO Recovery Based on differedupdate, Recovery technique based on immediate update, shadow paging,ARIES,Recovery in multidatabase system, Database backup and recovery from catastrophic Failures.

5: Distributed Database:[10L]

Principles of Distributed Databases, Framework for distribution, translation of global queries into fragment queries, query optimization and management of distributed transaction, concurrency control and reliability in distributed databases, Administration of distributed databases. Future Trends in data models: Semantic data models, DM for loosely structured data items, Multimedia database.

6: Emerging Technologies:[10L]

XML Databases: XML-Related Technologies-XML Schema- XML Query Languages- Storing XML in Databases-XML and SQL- Native XML Databases- Web Databases- Geographic Information Systems- Biological Data Management- Cloud Based Databases: Data Storage Systems on the Cloud- Cloud Storage Architectures-Cloud Data Models- Query Languages-Introduction to Big Data-Storage-Analysis.

7: NoSQL Database: [7L]

Introduction of NoSQLdatabases: Document Database,Graph Stores,Key value stores,Column stores, case study of Mongo DB(Document Based) and Neo 4J(Graph Based) databases

LIST OF BOOKS:

1. Fundamentals of Database Systems – By Elmasri & Navathe,Addison Wesl
2. Database System Concepts, Abraham Silberschatz, Henry F. Korth, Mc Graw Hill
3. J.D.Ullman Principles of Database Systems,Galgotia,New Delhi
4. S.Ceri and G.Relagatti Distributed Databases,McGraw Hills
5. C. Papadimitrious The Theory of Database Concurrency Control,Computer Science Press
6. T.Oszu& P.valduriez,Principles of Distributed Database Systems Pentice Hall.

Course Title: Software Architecture, Design Patterns and Frameworks Lab

Course Code: MIT 25

Marks: 50

Credits: 2

Course Pre-Requisites:

- Introductory course on Software Engineering

Course Objectives:

- Implement the various concepts of Object Orientation
- Implement the various Design Patterns
- Usage of various Architectural patterns and Frameworks.

Learning Outcomes:

- Understand various design patterns and their application in Software Development
- Understand software architecture and frameworks
- Understand antipatterns and steps that should not be taken while developing software

Syllabus:

This course will have programming assignments for the various types of patterns and frameworks discussed in the corresponding theory Course.

1. Implementation of various concepts of Object Orientation [4P]
2. Implementation of the 21 Design patterns (Creational, Structural and Behavioural) [8P]
3. Experiments on Architectural Patterns [9P]
4. Experiments on Enterprise Frameworks [9P]

Course Title: Design and Analysis of Algorithms Lab

Course Code: MIT 26

Marks: 50

Credits: 2

Course Pre-Requisites:

- Introductory course on Design and Analysis of Algorithms

Course Objectives:

- Understand the various algorithm design approach

Learning Outcomes:

- Implementation of various algorithmic approach.
- Implementation of algorithms for real life problem.

List of suggested Assignments:

1. Implementation of algorithms using divide and conquer approach. [5P]
    a. Binary Search
    b. Quick Sort
    c. Merge Sort

2. Implementaion of algorithms using dynamic programming approach. [5P]
    a. Assembly Line Scheduling
    b. Longest Common Subsequence
    c. Matrix Chain Multiplication
    d. Optimal Binary Search Tree
3. Implementaion of algorithms using Greedy programming approach. [5P]
    a. Huffman Codes
    b. Optimal Storage on Tapes
    c. Minimum Cost Spanning Tree(Prim’s and Kruskal Algorithm)
4. Implementaion of Backtracking programming approach for various problems. [5P]
    a. 8-Queen’s Problem
    b. Sum of Subsets
    c. Graph Coloring
5. Implementaion of various Graph algorithms. [5P]
    a. Dijikstra’s Algorithm
    b. Bellman Ford Algorithm
    c. Floyd Warshall Algorithm
6. Implementation of various internet algorithms. [5P]
    a. Tries
    b. Text Compression
    c. Text Similarity Testing

Course Title: Advanced Database Management Systems Lab

Course Code: MIT 27

Marks: 50

Credits: 2

Course Pre-Requisites:

- Introductory course on Database Management Systems

Course Objectives:

- Understand the concept of a database transaction and related database facilities, including concurrency control, backup and recovery, and data object locking and protocols.

Learning Outcomes:

- Implementation of different database architectures.
- Proficiency in storing, managing and interrogating complex data
- Representation the database using XML and work on it

List of suggested Assignments:

1: Revision and Normalization [3P]
2: Advance SQL- Dynamic SQL, Triggers, Assertions[3P]
3: Advance SQL- Stored Procedures[2P]
4: Indexing[1P]
5: Views,Roles,Grants[3P]
6: Data recovery techniques in databases. [2P]
7: Design XML Schema and perform queries using Xquery and Xpath[3P]
8: Introduction to IndexedDB[2P]
9: Introduction to NO SQL database [1P]
10: Creating Documents,Collection,inserting records,embedding[2P] documents
11: Querying the documents [2P]
13: Aggregation Framework [2P]
14: Sharding [2P]
15: Application development on No SQL Database using Java/PHP or equivalent. [2P]

Course Title: Data Mining

Course Code : MIT31

Marks: 100

Credits: 4

Course Pre-Requisites:

- An introductory course on DBMS

Course Objectives:

- Identify the key processes of data mining, data warehousing and knowledge discovery process
- Describe the basic principles and algorithms used in practical data mining and understand their strengths and weaknesses
- Apply data mining methodologies with information systems and generate results which can be immediately used for decision making in well-defined business problems

Learning Outcome:

- Understanding of various Data Mining Algorithms.
- Understanding various Data Mining tools such as Weka etc.

Syllabus:

1: Introduction and Background

[6L]

Introduction to the multidisciplinary field of data mining,. Discussion on the evolution of database technology that has led to the need for data warehousing and data mining. Stress on importance of its application potential. Introduction to the different key words and techniques

2: Data Warehousing And OLAP

[8L]

Insight of data warehouse and on-line analytical processing, Aggregation Operations, models for data Warehousing, star schema, fact and dimension tables Conceptualization of data warehouse and multidimensional databases. Life cycle of data warehouse development. Relationship between data warehouse and data mining

3: Data Mining Primitives

[8L]

Data preprocessing including data cleaning, data integration, data transformation. Definition and Specification of a generic data mining task. Description of Data mining query language with few example queries

4: Association Analysis

[9L]

Different methods(algorithms) for mining association rules in transaction based data bases. Illustration of confidence and support. Multidimensional and multilevel association rules. Classification of association rules. Discussion on few association rule algorithms e.g. Apriori, frequent pattern growth etc.

5: Classification and Predictions

[12L]

Different Classification algorithm, including C4.5, CART etc., use of genie index, decision tree induction, Bayesian classification, neural network technique of back propagation, fuzzy set theory and genetic algorithms

6: Clustering

[11L]

Partition based clustering, Hierarchical clustering, model based clustering for continuous and discrete data. Discussion on scalability of clustering algorithms

Web Mining: web usage mining, web content mining, web log attributes, use of web mining in efficient surfing and personalization.

Mining Complex type of data: Data mining issues in object oriented databases, spatial databases and multimedia data bases, time-series data bases, and text data bases

7: Applications of Data Warehousing And Data Mining

[6L]

Exploration of web sites on data ware housing and data mining applications including bibliography data bases, Corporate Houses and Research labs

LIST OF BOOKS:

1. Jiawei Han and Micheline Kamber, 2001, “Data Mining Concepts and Techniques,” Indian Reprint, Harcourt India Private Limited, ISBN 1-55860-489-8,1st Edition.
2. Vipin Kumar, Margaret Dunham, 2003, “Data Mining: Introductory and Advanced Topics,” Prentice Hall (Pearson Publication), ISBN 0-13-088892-3, 1st Edition.
3. Arun K Pujari, “Data Mining Techniques”. Universities Press
4. Pang-Ning Tan, Michael Steinbach,Vipin Kumar, Introduction to Data Mining, Pearson Education
5. Soumen Chakravarty, Web Mining

6. T. Mitchell, 1997, “Machine Learning”, McGraw Hill
7. S.M. Weiss and N. Indurkhya, 1998, “Predictive Data Mining”, Morgan Kaufmann
8. M. Jarke, M. Lenzerni, Y. Vassiliou, and P. Vassiladis, 2000, “Fundamentals of Data Warehouses”, Springer Verlag, Isbn 3-540-65365-1

Course Title: Information Retrieval

Course Code: MIT 32

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- The objective of the course is to introduce students to the theoretical underpinnings of information retrieval (IR), an active and rapid growing branch of applied computational science. Main topics of the course include document representation, document indexing, digital information storage, retrieval, and distribution. Emphasis is given to application of IR theories and practices to web indexing and web search engine

Learning Outcomes:

- Develop system for IR using various models
- Perform Query evaluation and Relevance feedback
- Design systems that included hyperlinks, multimedia and the web

Syllabus:

1.Overview of Information Retrieval

[3L]

Function of an IR system, Kinds of IR system, Components of an IR system, Problems in designing an IR system.

2.Boolean Model, Term vocabulary and Posting Lists [6L]

Term-Document Incidence matrix, Building an inverted index, Processing boolean queries, Obtaining character sequence in a document, Choosing a document unit, Tokenization, Stop word removal, Equivalence classing of terms, Stemming and Lemmatization, Porter's Algorithm for Stemming, Skip Pointers, Biword indexes, Positional indexes

3. Dictionaries and Tolerant Retrieval [6L]

Search structures for dictionaries, Wildcard queries, Permuterm indexes, k-gram indexes for wildcard queries, Spelling correction, computation of Levenshtein distance, k-gram indexes for spelling correction, Context-sensitive spelling correction, Phonetic correction, Soundex Algorithm

4. Index Construction and Compression [5L]

Blocked sort-based indexing, Single-pass in-memory indexing, Distributed and Dynamic indexing, Statistical properties of terms, Dictionary compression, Postings file compression

5. Vector Space and Probabilistic Models [6L]

Term frequency and Weighting, Inverse Document frequency, Computing Similarity Coefficient, Cosine Similarity between query and document vectors, Review of Probability theory, Ranking documents by using probabilistic retrieval

6. Evaluation, Relevance Feedback and Query Expansion [6L]

Standard test collections, Evaluation of unranked retrieval sets, Precision, Recall and F-measure, Assessing relevance, Kappa measure for inter-judge agreement, A/B testing, result snippets, relevance feedback and pseudo-relevance feedback, Rocchio algorithm, Global methods for query re-formulation, Query expansion and automatic thesaurus generation

7. XML Retrieval [7L]

Basic XML concepts, Challenges in XML retrieval, A Vector Space Model for XML retrieval, Evaluation of XML retrieval, Text-centric versus data-centric XML retrieval

8. Parallel and Distributed IR [6L]

Parallel Computing, Performance Measures, MIMD and SIMD architectures, Distributed Computing, Collection partitioning, source selection, Query processing, web issues

9. Multimedia IR [8L]

Multimedia data support in commercial DBMSs, MULTOS data model, Query

languages, request specification, conditions on multimedia data, uncertainty, proximity and weights in query expressions, spatial access methods, a generic multimedia indexing approach, one-dimensional time-series, two dimensional colour images, automatic feature extraction

10. IR and the World Wide Web [7L]

Background and history, web characteristics, web graph, search engine optimization, advertizing as the economic model, size of search engine index, sampling techniques, duplication, web crawling, crawling architecture modules, DNS resolution, URL frontier, anchor text and the web graph

List of Books

1. Christopher Manning, Prabhakar Raghavan and Hinrich Schutze: Introduction to Information Retrieval
2. D. Grossman and O. Frieder.2004, Information Retrieval: Algorithms and Heuristics, Kluwer, Second Edition
3. Ricardo Baeza-Yates and Berthier Ribeiro-Neto: Modern Information Retrieval
4. Korfhage, Robert R , 1997. Information Storage and Retrieval, 191-218. New York: Wiley,
5. C. J. van RIJSBERGEN, INFORMATION RETRIEVAL A ,Group, University of Glasgow (www.dcs.gla.ac.uk)
6. S. Chakrabarti. Morgan Kaufmann, 2002. Mining the Web: Analysis of Hypertext and Semi Structured Data.

Course Title: Data Mining & Information Retrieval LAB

Course Code: MIT 36

Marks: 50

Credits: 2

Course Pre-Requisites:

- None

Course Objectives:

- The objective of the course is to introduce students to the actual implementation of latest technologies that are used in the IT industry &

implementation of concepts in Information Retrieval and Data Mining Techniques.

Learning Outcomes:

- Knowledge of implementation of Data Mining and Information Retrieval concepts.

List of Experiments

1. Programs to implement Association Rule Mining Algorithms
[2P]

2. Programs to implement Clustering Algorithms
[5P]

3. Programs to implement Classification Algorithms
[4P]

4. Program to implement Sequential Pattern Matching
[3P]

5. Program to implement Temporal Pattern Mining
[3P]

6. Program to compare and contrast Association Measures
[1P]

7. Program to implement Boolean Model (Generate Term Incidence Matrix along with frequency)
[1P]

8. Program to implement stop word removal.
[1P]

9. Program to implement Porter's stemming algorithm.
[1P]

11. Program to implement Permuterm Indexes
[2P]

12. Program to implement Vector Space Model (Similarity Coefficient) [3P]

13. Program to implement Vector Space Model (Cosine Similarity) [2P]

14. Program to implement Probabilistic Model [2P]

Use of Apache Lucene and NLTK tools is recommended.

LIST OF ELECTIVES

1) Software Metrics & Project Management
2) Object Oriented Analysis and Design using UML
3) Mobile Computing
4) Introduction to Data Compression
5) Embedded Systems Design
6) Compiler Design
7) Computer Graphics
8) Natural Language Processing
9) Image Processing
10) Distributed Systems
11) Theory of Computation
12) Middleware Technology
13) Software Testing
14) Operating Systems, Kernel and Network Programming
15) Cloud Computing
16) Network Security

Course Title: Software Metrics & Project Management

Marks: 100

Credits: 4

Course Pre-Requisites:

- Introductory course on Software Engineering

Course Objectives:

- Provide a deeper understanding of various software metrics and project management concepts

Learning Outcomes:

- Understand the various types of management namely scope, time, cost, quality, human resource, communication, risk, procurement and integration management.
- Understand software metrics and quality standards.

Syllabus:

1. Introduction [3L]

Introduction to Project and Project management, Project phases and project life cycle, organizational structure, Qualities of Project Manager.

2. Project Management Components. [6L]

Project Integration Management-Project plan development and execution, change controls, configuration management.

3. Scope & Time Management [6L]

Strategic planning ,scope planning, definition ,verification and control, Activity planning, schedule development and control.

4. Cost & Quality Management [6L]

Cost estimation and Control, Quality planning and assurance.

5. Human Resource & Communication Management [9L]

Organizational planning, staff acquisition, Information distribution, reporting.

6. Risk Management [6L]

Risk identification, Quantification and control.

7. Procurement Management [4L]

Solicitation, contract administration.

8. Software Metrics [6L]

The scope of software metrics, software metrics data collection, analyzing software data, measuring size, structure, external attributes.

9. Software Reliability [4L]

Measurement and prediction, resource measurement, productivity, teams and tools.

10. Planning a measurement program. [6L]

Metrics plan: Developing goals, questions and metrics.
Where and When: Mapping measures to activities.
How: Measurement tools.
Who: Measurers , analyst, tools revision plans.

11. Quality Standards – CMM, PSP/TSP [4L]

List of Books:

1. Information Technology Project Management By -Kathy Schwalbe.
2. Software Metrics A rigorous and practical approach By – Norman Fenton, Shari Lawrence Pfleeger.
3: Software Engineering By- Roger Pressman.

Course Title: Object Oriented Analysis and Design using UML

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- Provide a deeper understanding of various software metrics and project management concepts

Learning Outcomes:

- To understand OOAD concepts in depth and be able to model a software system using the various modeling techniques

Syllabus:

1. Introduction [6 L]

Role of analysis and design in software development, purpose of analysis models, design models, introduction to OO analysis and design approach, comparison of OOAD and SSAD approaches, over view of OO analysis and design activities, overview of popular OOAD methodologies - common features and differences. Introduction to UML, Overview of various UML models.

2. Use case Modelling [6 L]

Concepts of use case model, actors, association of actors to use cases, use case description, structuring use cases with «include» and «extend» relationships, purpose of use case modelling, guidelines for use case modelling, finding actors, finding use cases, use case realization

3. Structural Modelling [6 L]

UML class diagram concepts - object, classes, class properties - attributes, operations, relationships between classes - association, aggregation, composition, inheritance, dependency, basic association adornments-multiplicity, role names; concept of package, grouping of classes into packages.

4. Behavioural Modelling [8 L]

Modelling object interaction using UML interaction diagrams - Sequence diagrams, collaboration diagrams; modelling the behaviour of reactive objects using UML state chart diagrams; modelling systems workflows or operations using UML activity diagram .

5. Introduction to the Unified Software Development Process [4 L]

Key features of the Unified Software Development Process - iterative model, use-case driven, architecture-centric, phases, iterations and workflows.

6. Software Architectural Design [8 L]

Architecture modelling in UML, Software architectural design issues, organization into subsystems, handling concurrency, allocation of subsystems to processors, choosing strategy for implementing persistent data stores, choosing strategy to control access to global resources, choosing strategy for software control implementation, overview of common architectural styles. Architecture modelling in UML -modelling active objects, deployment diagram.

7. Class design [8 L]

Class design activities, guidelines for - designing algorithms, design optimization, implementing state chart of class, adjustment of class hierarchies to increase reuse; design options for implementing associations, refinement of classes with attributes details and operations details, determining visibility between objects, physical packaging of classes into software modules, UML implementation diagram - Component diagram.

8. OO Design Patterns [10 L]

Introduction to OO design patterns, a template for describing designs patterns, a classification scheme for OO design patterns, uses of design patterns, patterns v/s frameworks, and illustration of some OO design patterns.

9.Implementation Guidelines [4 L]

Mapping a design class diagram for implementation in OO languages such C++ or Java. Overview of code generation and reverse engineering features of an OO case tool.

List of Books:

1. Craig Larman, Applying UML and Patterns: An Introduction to Object Oriented Analysis and Design; Pearson Low price edition
2. Martin Folwer and Kendall Scott; UML Distilled; Addison- Wesley
3. Object Oriented Analysis and Design using UML by Mahesh P. Matha, PHI Learning Pvt Ltd.
4. James Rumbaugh, Michael Blaha, William Premerlani, Frederick Edduy and William Lorensen; Object-Oriented Modelling and Design; Prentice-Hall
5. Grady Booch; Object-Oriented Analysis and Design with Applications,2nd Edition; Addison- Wesley
6. Jacobson, Booch and Rumbaugh; Unified Software Development Process; Addison- Wesley
7. Hans-Erik Eriksson and Magnus Penker; UML Toolkit; Wiley publishing
8. Rebecca Wirfs-Brock, Brian Wilkerson and Lauren Wiener; Designing Object-Oriented Software; Prentice Hall India

Course Title: Mobile Computing

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- To understand the basic concepts of Mobile Computing

Learning Outcomes:

- Apply data communicating methods and networking protocols for wireless and mobile environments.
- Understand positioning techniques and location based services and applications.
- Utilize and employ application frameworks for developing mobile applications.

Syllabus:

1. Introduction to Mobile Computing **[4L]**

i. Introduction and need for Mobile computing

ii. Mobility and portability
iii.Mobile and Wireless
devices iv.Applications
v. Brief History of wireless communication

2. Wireless Transmission [4L]

i. General Concepts of multiplexing and modulation
ii. Spread Spectrum
iii. Cellular Systems
iv. Cellular Phone Array
v. Mobile Phone Technologies (1G, 2G, 2.5G, 3G, 4G)

3. Medium Access Control Layer [4L]

i. Why specialized MAC? - hidden and exposed terminals- near and far terminals
ii.General Concepts and comparison of SDMA, FDMA, TDMA ,CDMA

4. Global System for Mobile Communication [10L]

i. Mobile Services (Bearer, Tele-and-supplementary services)

ii. System Architecture- Radio subsystem - Network and switching subsystem - Operation subsystem
iii. Protocols - Localization and calling - Handover
iv. Value Added Services- SMS Architecture, Mobile Originated and Mobile

Terminated ,procedures - Cell Broadcast Service Architecture, Message Transfer Procedure – MMS Architecture, Protocol framework, Message Transfer

Procedure - Location Services, Logical Reference Model, Control Procedures, Network Architecture, determination of Location Information, Location based services
v. GPRS

5. Mobile IP [6L]

i. Goals, assumptions and requirements
ii. Entities and terminologies
iii. Agent Discovery
iv. Registration
v. Tunnelling and encapsulation
vi. Reverse Tunnelling
vii. IPv6
viii. IP micro-mobility support – Cellular IP, Hawaii, Hierarchical mobile IPv6

ix. Mobile Routing : Destination sequence distance Vector, Dynamic Source Routing, Alternative Matrix, Ad hoc Routing Protocols -Flat, Hierarchical, Geographic-position-assisted

6. Mobile TCP [6L]

i. Traditional TCP - Congestion Control, Slow start, Fast retransmit / Fast recovery
- Implications on mobility

ii. Classical TCP improvements Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit / Fast recovery, Transmission / Timeout freezing, Selective Retransmission, Transaction oriented TCP

iii. TCP over 2.5/3G wireless networks

7. Wireless Application Protocol **[6L]**

i. Architecture
ii. Wireless datagram protocol
iii. Wireless transport layer security
iv. Wireless transaction protocol
v. Wireless session protocol
vi. Wireless application environment
vii. WML
viii. WML Scripts
ix. Push Architecture
x. Push – Pull Services

8. Platform/Operating Systems **[4L]**

i. Palm OS
ii. Windows CE
iii. Embedded Linux
iv. J2ME (Introduction)
v. Symbian (Introduction)
vi. File Systems (Book1)

9. Java for Wireless Devices **[2L]**

i. Setting up the development environment
ii. Basic Data types, Libraries (CLDC, MIDP)

10. UI Controls **[4L]**

i. Displayable and Display
- Image
- Events and Event Handling
- List and choice
- Text box
- Alerts

11. Persistent Storage **[2L]**
i. Record Stores
ii. Records
iii. Record Enumeration

12. Network MIDlets **[4L]**
i. The Connection Framework
ii. Connection Interface
iii. Making a connection using HTTP
iv. Using datagram connection

13. Wireless Messaging **[4L]**

i. Architecture for Messaging application
ii. Messaging API
iii. Types of applications

iv. Pros and cons of messaging

List of web reference:

1. http://java.sun.com/products/wma
2. http://forum.nokia.com

List of Books:

1. Mobile Communications Jochen Schiller, Pearson Education, 2nd Edition

2. Pervasive Computing Technology and Architecture of Mobile Internet Applications

JochenBurkhardt, Dr. Horst Henn, Steffen Hepper, Klaus Rintdorff, Thomas Schack , Pearson Education

3. Wireless Java Programming with J2ME Yu Feng and Dr, Jun Zhu ,Techmedia Publications, 1st edition .

4. Mobile Networks GSM and HSCSD NishitNarang, SumitKasera, TataMcGrawHill

5. Mobile Computing Asoke K Talukdar, Roopa R. Yavagal, TataMcGrawHill

Course Title: Introduction to Data Compression

Max Marks:100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- Provide a deeper understanding of Data Compression

Learning Outcome:

- Implementation of Data Compression techniques in real life applications.

Course Contents:

1. Introduction [5L]

Compression techniques, Modelling and Coding

2. Huffman Coding [6L]

Huffman Coding algorithm, Adaptive Huffman Coding, Golomb Codes, Rice Codes, Tunstall Codes, Applications of Huffman Coding

3. Arithmetic Coding **[7L]**

Coding a sequence, Generating a binary code, Comparison of Huffman and Arithmetic Coding, Adaptive Arithmetic Coding, Applications

4. Dictionary Techniques **[6L]**

Static Dictionary, Adaptive Dictionary, Applications

5. Context based Compression **[8L]**

Prediction with partial match, the Burrows- Wheeler Transform, Associative code of Buyanovsky, Dynamic Markov Compression

6. Lossless Image Compression **[10L]**

The old JPEG standard, CALIC, JPEG – LS, Multi resolution approaches, FACSIMILE Encoding, MRC – T.44

7. Scalar Quantization **[9L]**

The Quantization problem, Uniform Quantizer, Adaptive Quantization, Non Uniform Quantization, Entropy coded Quantization

8. Vector Quantization **[9L]**

Advantages of Vector quantization over Scalar quantization, The Linde-Buzo-Grey Algorithm, Tree Structured Vector Quantizers, Structured Vector Quantizers, Variations on the Theme.

List of Books:

1. Khalid Sayood, Introduction to Data Compression, Morgan Kauffmann

Course Title: Embedded Systems Design

Marks: 100

Credits: 4

Course Pre-Requisites:

- Introductory course on Operating Systems

Course Objectives:

- Understand the Embedded Systems architecture, process and development
- Learn the basic concept of RTOS

Learning Outcomes:

- Design, test and critically evaluate embedded solutions to real world situations.
- Recognize the key features of embedded systems in terms of computer hardware and be able to discuss their functions.

Syllabus:

1. Introduction to embedded system [4L]

A First Look at Embedded systems- Examples of Embedded systems- applications area-categories of embedded system – recent trends in embedded system

2. Design challenge [4L]

Optimizing design metrics Common design metrics- Processor technology- General-purpose processors – software - Single-purpose processors – hardware- Application-specific processor- IC technology

3. Architecture of embedded system [4L]

Hardware architecture –software architecture - Programming for embedded system.

4. The process of embedded system development [5L]

Interrupts-Microprocessor-Architecture-Interrupt Basics-The Shared-Data Problem-Interrupt Latency shared data problems- survey of software architecture -Round-Robin-Round-Robin with Interrupts- Interrupt Latency – RTOS.

5. RTOS [6L]

Architecture of kernel –task and task scheduler-interrupt service routines-semaphores-message queues- mail boxes-pipes –events-timer – memory management –interrupt routines in RTOS Environment overview of embedded /real time operating system

6. Embedded Software Development Tools [4L]

Host and Target Machines-Linker/Locators for Embedded Software-Getting Embedded Software into the Target System.

7. Debugging Techniques [4L]

Testing on Your Host Machine-Instruction Set Simulators-The assert Macro-Using Laboratory Tools

8. An Example System [3L]

What the Program Does-Environment in which the Program Operates

9. Task Image creation [5L]

Operating system software –target image creation for windows XP embedded-porting RTOS on a microcontroller based development board.

10. Representative embedded systems [5L]

Programming in Linux-programming in RTLinux-Development of Navigation System –Development of protocol converter-mobile Java applications.

11. State machine and concurrent process models [12L]

Introduction- Models vs. languages, text vs. graphics- Models vs. languages-Textual languages versus graphical languages-An introductory example- A basic state machine model: finite-state machines (FSM)-

Finite-state machines with data path model: FSMD- Using state machines-Describing a system as a state machine-Comparing the state machine and sequential program model- Capturing a state machine model in a sequential programming language-Hierarchical/Concurrent state machine model (HCFSM) and the State charts language

12. Program-state machine model (PSM) [4L]
The role of an appropriate model and language

List of Books:
1. Embedded software primer by David Simon – Pearson
2. Art of embedded system by Jack Ganssle
3. Embedded systems Architecture by Tammy Noergaard – Elsevier publications
4. Embedded /Real time systems – by DR.K.V.K.K.Prasad. – Dreamtech
5. Esterel language by Gerard Berry (web site reference)
6. Embedded system design by Arnold S.Berger
7. model checking by Edmund M.clark
8. Embedded Systems Building Blocks by Jean LaBrosse
9. Embedded Systems Design by Arnold Berger
10. The Art of Programming Embedded Systems by Jack Ganssle

Course Title: Compiler Design

Marks: 100

Credits: 4

Course Pre-Requisites:

- An introductory course on theory of computation

Course Objectives:

- To enable the student to understand compiler construction and equip them with skills to write a compiler for a programming language.

Learning Outcomes:

- Understanding of the layers of the compiler design.
- Various tools such as Lex etc.

Syllabus:

1: Introduction [6L]

Lexical analysis, Regular Expressions, Finite automation. N.F.A., N.F.A. to D.F.A. conversion, D.F.A., minimization of D.F.A., Lex tool

2: Context Free Grammar [9L]

Derivations & Parse trees, Syntax analysis: Parsing, Top Down Parser, Recursive descent Parser, Predictive parsing, LL(1) Parsing table, Bottom Up Parsing, Shift Reduce parsing, Operator precedence parsing, LR Parsing methods, SLR, LRDL, LALR, YACC tool

3: Syntax Directed Translation [9L]

Syntax directed translation schemes, Implementation of syntax directed translation schemes, Intermediate codes, Post fix notation parse trees & syntax trees, three address codes, quadruples, triples, Translation of assignment statements, Boolean expression, statements that after flow of control, Post fix translation, Translation with Up down parsing

4: Error Detection & Recovery [7L]

Errors, lexical phase errors, Syntactic phase errors, semantic errors

5: Code Optimization [8L]

Loop optimization, DAG representation of basic block, value numbers & algebraic laws, Global data flow analysis, Dominators, Reducible flow graph, Depth first search, Loop invariant computation, Induction variable elimination

6: Data Flow Analysis [7L]

Reaching definition, Available Expression , copy propagation, Backward flow problems, Very busy, expression & code hoisting code

7: Code Generation [7L]

A simple code generation, code generation from DAG & labeled trees

8: Register Allocation [7L]

Coloring by implication, coalescing, graph coloring implementation, Register allocation for Trees

List of Books

1. Alfred V. Aho, Jeffreys D. Ullman &Ravi Sethi, “Principles of Computer Design“ Narosa Publication
2. Trembley et al, “ Theory & Practice of Compiler Writing “, McGraw Publication
3. Andrew W. Appel, “ Modern Compilers Implementation in Java”
4. Dhamdhere “ Compiler Principles”

Course Title: Computer Graphics

Marks:100

Credits:4

Course Pre-Requisites:

- An introductory course on Data Structures and Algorithms

Course Objectives:

- To understand the concepts of Graphic Algorithms, Geometrical transformations and Modeling

Learning Outcome:

- Describe the purpose of Computer Graphics and its applications
- Describe and implement methods for performing 2-Dimentional geometric transformations
- Describe the concept of 3-Dimentional Graphics and methods for performing 3-Dimensional geometric transformations.
- Discuss basic illumination models and surface rendering algorithms.

Course Contents:

1: Basic background **[5L]**

2D and 3D Transformations, Cyrus-Beck line clipping algorithm, Polygon clipping.

2: 3D Viewing **[5L]**

Viewing pipeline, Parallel and Perspective projections, view volumes, clipping

3: Representing Curves and Surfaces **[6L]**

Parametric, curves, continuity conditions, cubic splines, Hermite interpolation, Bezier curves and surfaces, B-spline Curves- uniform nonrational, cubic periodic, open uniform, uniform,non uniform rational types (NURBS), Subdividing curves, Displaying spline curves using forward difference scheme, parametric bicubic surfaces.

4: Solid Modeling **[7L]**

Sweep representation, Constructive solid geometry methods, representation through Octrees, Binary Space Partitioning trees.

5:Visible Surface Determination **[8L]**

Issues in Visible surface determination Coherence, perspective view, extents and bounding volume, backface culling, Z-Buffer and A-Buffer Algorithms, use of

Binary Space Partitioning trees, representing 3D data using Octrees, Boolean operations on Octrees, marching cubes, Visible surface ray tracing.

6: Illumination Models & Rendering [7L]

Diffuse and Specular illumination model, reflection vector computation, Shading models for polygons – polygon mesh shading, Gouraud and Phong Shading, problems with interpolated shading, Bumpd mapping, Transparency, shadows, Ray tracing.

7: Introduction to Animation [6L]

Perception, Animation production, use in film and videos, orientation representation andinterpolation – Euler angle representation, motion display considerations.

8: Animation – Low Level Control [8L]

Motion along a curve – computing are length, speed control – sine interpolation User specifieddistance time functions, path following, key-frame systems – shape interpolation, free-formdeformations, Morphing – 2D object warping.

9: Animation – High Level Control [8L]

Hierarchical modeling and Kinematics – inverse kinematics, Jacobian, rigid body simulation,collision detection, Particle systems – particle generation, attributes, termination, rendering,Flocking behavior – interacting with other members, leader, collision avoidance, modelingwater, fire, explosions, waves, clouds.

List of Books

1. Foley, Van Dam, Feiner, Hughes, Computer Graphics – Principles and Practices , Addison Wesley.

2. Rick Parent, "Computer Animation: Algorithms and Techniques, Morgan-Kaufman,

3. Hearn & Baker, Computer Graphics, Prentice Hall of India.

Course Title: Natural Language Processing

Marks:100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- To study fundamental concepts of Natural Language Processing and to introduce the basics of Language processing from algorithmic viewpoint.

Learning Outcome:

- Compose key NLP elements to develop higher level processing chains
- Assess / Evaluate NLP based systems
- Choose appropriate solutions for solving typical NLP subproblems (tokenizing, tagging, parsing)

Course Contents:

1. **Introduction** **[4L]**
   Ambiguity, Models and algorithm, Language, thought and understanding

2. **Regular Expressions and Automata** **[7L]**
   Regular Expressions, Basic Regular Expression Patterns, Disjunction, Grouping, and Precedence, Advanced Operators, Regular Expression substitution, Memory, and ELIZA

3. **Finite-State Automata** **[4L]**
   Using an FSA for Recognition, Formal Languages, Non-Deterministic FSAs, Using an NFSA to Accept Strings, Recognition as Search

4. **Morphology and Finite-State Transducers** **[8L]**
   English Morphology, Inflectional Morphology, Derivational Morphology, Finite-State Morphological,Parsing, The Lexicon and Morphotactics, Morphological Parsing with Finite-State Transducers, Orthographic Rules and Finite-state Transducers, Combining FST Lexicon and Rules, Lexicon-free FSTs: The Porter Stemmer

5. **N-grams** **[7L]**
   Counting Words, Simple N-grams, Smoothing, Backoff, Deleted Interpolation, N-grams for Spellings, Entropy

6. **Word Classes and Part-of-Speech Tagging** **[5L]**
   English classes, tagsets POS tagging, Rule based POS tagging, Stochastic POS tagging, HMM tagging, Transformation based tagging, Multiple tags and multiple words, unknown words

7. **Features and Unification** **[4L]**
   Feature structures, Unification of feature structures, Feature structures in grammar, Implementing unification, Parsing with unification constraints, Types and inheritances

8. **Lexicalized and Probabilistic Parsing** **[3L]**
   Probabilistic Context free grammars. Problems with PCFGs, Probabilistic Lexicalized CFGs, Dependency Grammars, Human Parsing

9. **Representing Meaning** **[7L]**

Computational Desiderata for representation, Meaning Structure of Language, Some Linguistically relevant concepts: Categories, Events, Representing time, Aspects, Representing beliefs, Pitfalls. Alternative approaches to meaning

10. Semantic Analysis **[8L]**

Syntax-Driven Semantic Analysis, Semantic Augmentations to context-Free Grammar Rules, Quantifier Scoping and the Translation of Complex-Terms. Attachments for a Fragment of English. Sentences, Noun Phrases, Verb Phrases. Prepositional Phrases, Integrating Semantic Analysis into the Earley Parser. Idioms and Compositionality, Robust Semantic Analysis

11. Lexical Semantics **[3L]**

Relation among lexes and their senses, WordNet, Internal structure of Words, Creativity and the Lexicon

List of Books:

1. Natural Language processing by Daniel Jurafsky, James H Martin, Pearson Education Asia

2. Jurafsky, Dan and Martin, James, Speech and Language Processing, Second Edition, Prentice Hall, 2008

Course Title: Image Processing

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- To understand the basic image processing operations.

Learning Outcomes:

- Explain how digital images are represented and manipulated in a computer, including reading and writing from storage, and displaying.
- Analyze and implement image processing algorithms

Syllabus:

1. Introduction **[4L]**

Image formation model, representation, spatial and Gray Level resolution, Colour models-RGB, CMY and HIS models

2. **Image Enhancement In Spatial Domain** **[8L]**
Piecewise linear transformation, Histogram equalization, Histogram specification, image averaging, spatial filters – smoothing and sharpening, Laplacian filter, sobel operator, Canny edge detector

3. **Image Enhancement In Frequency Domain** **[8L]**
2D Discrete Fourier transform and its inverse, filtering in frequency domain, Ideal and Gaussian Low pass filters, high pass filtering, separability property 0f 2D Fourier transform, Fast Fourier Transform

4. **Image Segmentation** **[6L]**
Line detection, Edge detection, Edge linking and boundary detection, Hough Transform, Thresholding, Region based segmentation

5. **Morphological Image Processing** **[8L]**
Logic operations involving binary images, Dilation and Erosion, Opening and closing, Applications to Boundary extraction, region filling, connected component extraction

6. **Image Compression** **[10L]**
Coding redundancy- Huffman coding, LZW coding, run length coding, Lossy compression – Lossy predictive coding, transform coding- DCT, bit allocation, Compression standards – JPEG, video Compression

7. **Image Representation** **[6L]**
Boundary description, Shape numbers, Fourier descriptors, Texture, principal Components based description

8. **3D Vision** **[10L]**
Projective geometry, single perspective camera, stereopsis, the fundamental matrix – its estimation from image point correspondences, applications of epipolar geometry in vision, correlation based and feature based stereo correspondence, shape from motion, optical flow

List of Books:

1. Gonzalez and Woods, "Digital Image Processing"' 2002, Pearson education, Asia
2. Sonka, Hlavac and Boyle Brooks/Cole, "Image Processing, Analysis, and Machine Vision", 1999, Thomson Asia Pte Ltd Singapore
3. Jain and Rangachar, "Machine Vision", 1999, McGraw Hill International Edition
4. Schalkoff, John Wiley and Sons, "Digital Image Processing & Computer Vision", 1989, John Wiley and Sons

Course Title: Distributed Systems

Marks: 100

Credits: 4

Course Pre-Requisites:

- **None**

Course Objectives:

- To introduce basic principles and foundations of distributed systems.

Learning outcomes:

- Distinguish the theoretical and conceptual foundations of distributed computing.
- Recognize the feasibilities and the impossibilities in managing resources.
- Identify the problems in developing distributed applications.

Syllabus:

1. Introduction [12L]

Examples of Distributed Systems, Resource Sharing and the Web, Challenges System Models, Introduction to Architectural Models, Functional Models, Characterization of Distributed Systems, Client-Server Communication, Distributed Objects and Remote Invocation, Communication Between Distributed Objects, Remote Procedure Call, Events and Notifications.

2. Distributed Operating Systems [18L]

Introduction, Issues, Communication Primitives, Inherent Limitations, Lamport's Logical Clock; Vector Clock; Causal Ordering; Global State; Cuts; Termination Detection. Distributed Mutual Exclusion, Non Token Based Algorithms, Lamport's Algorithm Token, Based Algorithms, Suzuki Kasami's Broadcast Algorithm, Distributed Deadlock Detection, Issues, Centralized, Agreement, Protocols, Classification, Solutions Applications.

3. Distributed Resource Management [10L]

Distributed File systems, Architecture, Mechanisms, Design Issues, Distributed Shared Memory, Architecture, Algorithm, Protocols, Design Issues. Distributed, Scheduling, Issues, Components, Algorithms.

4. Distributed Algorithms [12L] Introduction to Distributed Algorithms, Kinds of Distributed Algorithm, Timing Models. Synchronous Network Algorithms: Synchronous Network Model, Leader Election in a synchronous Ring, Algorithms in a General

Synchronous Networks, Distributed Consensus with Link Failures, Distributed Consensus with Process failures, More Consensus problems.

5. Resource Security and Protection **[8L]**

Introduction, The Access Matrix Model, Implementation of Access Matrix Model, Safety in the Access Matrix Model, Advanced Models of protection, Data Security.

List of Books:

1. George Coulouris, Jean Dellimore and Tim KIndberg, “Distributed Systems Concepts and Design”, Pearson Education, 4th Edition, 2005.

2. MukeshSinghal and N. G. Shivaratri, “Advanced Concepts in Operating Systems”, McGraw Hill, 2001.

3. Joshy Joseph and Craig Fellenstein, “Grid Computing”, IBM Press, 2004.

4. Ajay D. Kshemkalyani and MukeshSinghal, “ Distributed Computing Principles,
Algorithms and Systems”, Cambridge University Press, 2008.

5. Pradeep K. Sinha, Distributed Operating Systems, PHI, 2005.

6. Nancy A. Lynch, Distributed Algorithms, Morgan Kaufmann Publishers, 2000.

Course Title: Theory of Computation

Marks:100

Credits:4

Course Pre-Requisites:

- None

Course Objectives:

- To introduce the student to the concepts of theory of computation in computer science. The student should acquire insights into the relationship amongst formal languages, formal grammars and automata.

Learning Outcome:

- Logic and set theory, functions and relations, formal languages and grammars.
- Finite-state automata, pushdown automata
- Turing machines, Church's Thesis, undecidability
- Recursively Enumerable Languages and Unsolvable Problems.

1: Introduction **[5L]**

Sets, Logic, Functions, Relations, Languages, Proofs, Mathematical Induction, Recursive definitions, Structural Inductions

2: Regular Languages and Finite Automata **[7L]**

Regular Languages and Regular Expressions, The memory required to recognize a language Finite Automata (DFA),Distinguishing one string from another, Union, Intersection, and Complement.

3: Nondeterministic and Kleene's theorem **[7L]**

NFA, Converting N**FA to DFA,€**-NFA ,Kleene's th**eorem, Converting an €**-NFA to an NFA Regular Languages, MyHill-Nerode theorem, Minimal finite Automata, The pumping lemma for regular languages, Closure properties, Decision Problem, Moore and Mealy Machine.

4: Context –free Grammars and Push down Automata **[7L]**

Context –Free Grammars and Languages, Derivation Trees and Ambiguity, An unambiguous CFG for algebraic Expression, Simplified forms and Normal Forms CNF, GNF Pumping Lemma, Closure Properties.

5: Push Down Automata **[7L]**

DPDA,PDA corresponding to a given CFG – Top-down PDA, Bottom-up PDA CFG corresponding to a given PDA, Closure properties of CFG.

6: Turing Machine and their languages **[12L]**

Turing Machine Introduction, Computing a Partial function with a Turing machine Combining Turing machine, Variations of Turing Machine, Nondeterministic Turing Machine Universal Turing Machine, Church-Turing Thesis

7: Recursively Enumerable Languages **[8L]**

Recursively Enumerable and Recursive, Enumerating a Language, General Grammars Unrestricted Grammars and Turing Machine, Context-Sensitive Language and Grammar Linear Bounded Automata, Chomsky Hierarchy

8: Unsolvable Problems **[7L]**

A non recursive language and unsolvable Decision problems, Reducing one problem to another ,The halting problem, Rice's Theorem, Closure Properties of families of language

List of Books:

1. Introduction to languages and the theory of computation, By John C. Martin, Tata McGraw Hill
2. Introduction to Automata Theory, Languages and Computation - By Hopcraft and Ullman, Narosa Publishing House.
3. Theoretical Science - By Krishnamurthy, AWEP.
4. Theory of Computer Science - By Brady, McGraw Hill.
5. Computations, Finite and Infinite Machines - By Minsky, Prentice Hall

Course Title: Middleware Technology

Marks: 100

Credits: 4

Course Pre-Requisites:

- Fundamental knowledge of Distributed Systems and knowledge of Java

Course Objectives:

- Understanding the characteristics of distributed systems, Asynchronous communication and Event based systems.
- Understanding of J2EE and Web services.

Learning Outcomes:

- Understand the distributed systems, asynchronous communication and event-based systems in detail.
- Gain knowledge of Servlet technology and Enterprise Java beans
- Understand web services and reflective middleware

Syllabus:

1. **Fundamentals of middleware** **[5L]**

   Introduction to middleware, MW definition, styles of MW, key players;

2. **Distributed systems characteristics** **[12L]**

   System models-architectural and fundamental models. RPC, Distributed objects-RNI, .NET Remoting, Name services-DNS, Time and global states, synchronization, Coordination and agreement, distributed transactions and recovery, Consistency & Replication, Fault Tolerance, Security

3. **Asynchronous communication and Event based systems** **[7L]**

   Notifications, message Queuing systems, peer to peer systems

4. **Middleware and enterprise services in J2EE** **[15L]**

   Servlets and EJBs.

5. **SOA & Web services** **[11L]**

   XML,SOAP, WSDL, UDDI & other protocols;

6. **Reflective middleware** **[10L]**

   Introduction to reflective middleware,Middleware oriented architectural patterns for enterprise systems.

LIST OF BOOKS:

Main Reading

1. Distributed Systems- Concepts and Design: George Couloris, Jean Dollimore, Tim Kindberg, Tim Kindberg, Third Edition, Pearson Education, Addison Wesley

Course Title: Software Testing

Marks: 100

Credits: 4

Course Pre-Requisites:

- Knowledge of analysis, design and programming

Course Objectives:

- To provide a detailed study of testing software and automated tools.

Learning Outcomes:

- Revise fundamentals of testing and learn about Functional testing and Object Oriented testing methods.
- Gain knowledge of test case design, execution and report
- Understand testing of web applications and automated testing tools

Syllabus:

1. Testing fundamentals **[6L]**

Software testing – Levels of software testing – Test activities – Testing Life Cycle – Test Organization – White Box testing – Basis Path Testing – Control Structure testing – Black Box Testing.

2. Functional testing **[11L]**

Equivalence Class Partitioning - Boundary Value Analysis – Cause-effect Graphing – Special cases. Performance Testing – Stress testing – Configuration Testing – Security Testing – Recovery Testing – Integration Testing – Regression Testing – Acceptance Testing.

3. Object oriented testing methods **[12L]**

Testing Methods at Class level – Interclass test case design- Testing for Specific Environment, architecture, and application - Testing patterns.

4. Testing process **[11L]**

Comparison of different techniques- Test Plan – Test case Design Procedure Specification – Test Case Execution and Analysis - Test Documentation - Reporting test results - Final test report Test Driven Development & Refactoring

5. Testing Web Application **[12L]**

Testing concepts for web apps – Content Testing – User Interface Testing – Component Level Testing – Navigation Testing – Configuration Testing – Security Testing – Performance Testing.

6. Testing Tools **[8L]**

Need for automated testing tools - Selection of testing tool – Tools used at various phases.

LIST OF BOOKS

1. Srinivasan Desikan, Gopalswamy Ramesh , “Software Testing : Principles and Practices”,Pearson Education, 2006

2. Software Testing in the Real World, by E. Kit (1995)

3. The Web Testing Handbook, by S. Splaine and S. Jaskiel

4. Testing Applications on the Web, by H. Nguyen, R. Johnson, and M. Hackett

5. Software Testing and Continuous Quality Improvement, by W. Lewis, et al

6. How to Break Software Security, by J. Whittaker, et al

7. Web resources: http://www.softwareqatest.com

Course Title: Operating Systems, Kernel and Network Programming

Marks: 100

Credits: 4

Course Pre-Requisites:

- An introductory course on Operating Systems

Course Objectives:

- To introduce the student to Operating Systems, Kernel and Network Programming

Learning Outcomes:

- Explain Unix file system including advanced file processing.

- Understand the key communication protocols that support the Network.
- Have a detailed knowledge of the TCP/UDP Sockets

Syllabus:

1. **Introduction to UNIX OS** **[14L]**
   Organization of unix user interface. Programmer interfaces. The environment of a unix process, system calls, programming system calls. File I/O, File and Directories, standard I/O library, File related system calls. Process control, Process relationships. Process groups, sessions, controlling Terminal, Process related system calls.
   Signals, Signal concept, Reliable and unreliable signals, Signal sets, Signal related system calls.
   Terminal I/O multiplexing, Memory mapped I/O, related system calls
   Interprocess Communication, Pipes, Message queues,Semaphores and shared memory. .
   Advanced interprocess communication, stream pipes, open server

2. **File I/O and Directories** **[7L]**
   File descriptor and basic file I/O calls. Duplicating file descriptors. File Types, File accesspermissions, Set-user-id and set-group-id bits. Setting file permissions. Changing file ownership.Soft and hardlinks.Reading directories.Synchronising file contents.Standard I/O library.

3. **Process** **[6L]**
   Environment of UNIX process.Command Line arguments.Environment variables.Memory allocation. Process relationship, Process groups, sessions, Controlling Terminal, Process related system calls. Foreground, Background Processes and Job control.Orphaned process groups.

4. **Signals** **[4L]**
   Signal concept, Reliable and unreliable signals, Signal sets, Signal related system calls. Non local jumps.Job control using signals.

5. **Terminal I/O** **[5L]**
   Special Input Characters.Canonical and Non canonical modes. Terminal Option flags. Getting and setting terminal attributes.Pseudo terminals. Opening and using pseudo Terminals.

6. **Advanced I/O** **[5L]**
   Nonbloking I/O, Record locking. Stream, I/O multiplexing, Memory mapped I/O, Asynchronous I/O.

7. **Inter-process communication** **[3L]**
   Pipes, Message queues, Semaphores and shared memory.

8. **Introduction to Computer Networks** **[4L]**
   Overview of computer networks, seven-layer architecture. Communication protocols, Internet protocols, SNA, Protocol comparison. Test networks and hosts, discovering network topology, 64 bit architecture

9. Transport Layer [5L]

Introduction to TCP and UDP protocols, TCP port numbers and concurrent servers, protocol usage by common internet applications

Berkeley Sockets: Socket address structures, socket functions for TCP Client /server, Day time client /server example

10. I/O Multiplexing [3L]

Basic I/O model, asynchronous and synchronous I/O model

11. Socket Options [4L]

Generic ,IPv4 , IPv6 and TCP.
Elementary UDP socket programming: Name and address conversions

List of Books:

1. Steven W.R. "Advanced Programming in UNIX Environment" - Pearson Education
2. Steven W.R., UNIX Network Programming, Second Edition, Pearson education
3. Ralph Davis, Win 32 network programming, Addison Wesley

Course Title: Cloud Computing

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- To study important approaches in the field of Cloud Computing.

Learning Outcomes:

- Understand Cloud infrastructure models and grid computing
- Gain knowledge about Google App Engine and programming it with Python.
- Have knowledge about Amazon cloud, Windows Azure Platform and Security in cloud computing

Syllabus:

1. Introduction to cloud computing [6L]

How it all began – Grid computing –utility computing-automatic computing – dynamic data centre alliance-hosting/outsourcing . Cloud computing defined – the SPL frame work for cloud computing –traditional software model – virtualization –the cloud services delivery model-cloud deployment model – key drivers to adopting the cloud –the impact of cloud computing on users-governance in the cloud-the barrier to cloud computing adoption in the enterprise.

2. Developing cloud services [8L]

Cloud infrastructure models-types of cloud service development-softwar as a service-platform as a service – infrastructure as a service-web services – on-demand computing –discovering cloud services deployment services and tools – amazon EC2 –google app engine – Microsoft azure.

3. Google app engine [8L]

The java runtime environment-the python runtime environment – the data store-development workflow-Setting up a google app engine account –setting development enrvironment –starting to program in phython with app engine.

4. Programming google app engine with python [8L]

A first real cloud application –the basic example –chat application-the basics of HTTP –mapping chat into HTTP.

5. Programming google app engine with java [8L]

Google app engine and java-managing server side data – building user interface in java – building the server side of java application.

6. Amazon cloud computing [7L]

Amazon s3-amazon EC2-the simple storage service-simple queuing services

7. Window azure platform [7L]

Windows azure-SQL azur-windows azure app facric- additional online services

8. Security [8L]

Data security –network security-host security –compromise –response.

List of Books

1. Geroge Reese ,cloud computing architecture ,O’Reilly publications.

2. Michael miller , cloud computing :web based application applications that change the way.
3. Tim mather, subra kumarswamryand sharhed Latif, cloud computing security and privacy, O’Reilly publication.
4. Mark C. chu carrol – code in the cloud –programming google app engine.
5. Using google app engine by Charles severance , O’Reilly publication.

Course Title: Network Security

Marks: 100

Credits: 4

Course Pre-Requisites:

- None

Course Objectives:

- To understand the concepts and theory of computer network security.

Learning Outcomes:

- Understand basics of Cryptography and security
- Gain knowledge about Block and Stream Ciphers, public key cryptography and asymmetric algorithms
- Have knowledge about authentication and web security protocols

Syllabus:

1. Foundations of Cryptography and Security [5L]

Ciphers and Secret Messages, Security Attacks and Services. Classical encryption techniques.

2. Mathematical Tools for Cryptography [6L]

Substitutions and Permutations, Modular Arithmetic, Euclid’s Algorithm, Finite Fields, Polynomial Arithmetic.

3. Design Principal of Block Ciphers [10L]

Theory of Block ciphers, Feistel Cipher network Structures, DES and triple DES, Modes of Operation (ECB, CBC, OFB, CFB), Strength of DES., AES

4. Pseudo Random Numbers and Stream Ciphers [9L]

Pseudo random sequences, Liner Congruential generators, Cryptographic generators, Design of stream Ciphers, RC4.

5. Public Key Cryptography [6L]

Prime Numbers and testing for primality. Factoring large numbers, Discrete Logarithms.

6. Asymmetric Algorithms [5L]

RSA, Diffie-Hellman, ElGamal, Introduction of Ecliptics curve cryptosystems, Key Management, Key exchange algorithms, Public Key Cryptography Standards.

7. Hashes and Message Digests [5L]

Message Authentication, MD5, SHA-3, HMAC

8. Digital Signatures, Certificate and Standards [4L]

Digital signature standards (DSS and DSA), Public Key Infrastructures, Digital certificates and Basics of PKCS standards.

9.Authentication [5L]

Kerberos , X509 Authentication Service

10. Web Security protocols [3L]

IP Security, Transport Layer Security (TLS)., Wireless Security,

11. System Security [2L]

Intrusion detection , Password management. Firewalls management

List of Books

Main Reading

1. Stallings William, " Cryptography and Network Security: Principles and Practises", 5th edition, Prentice Hall
2. Kahate Atul, "Cryptography and Network Security" Tata McGraw-Hill.

Supplementary Reading

1. Menezes A. J., P.C. Van Oorschot and S.A. Vanstone, "Handbook of Applied Cryptrography"