Course Name: Software Construction

Course Code: MCA 221

Objectives:

To prepare the students to become good software coders, designers as well as Managers It will help the students understand and learn powerful software development techniques.

Prerequisites:

Fundamentals of programming

Contents:

1. Software Construction [10%]
Introduction, Metaphors for Software Development, Upstream Prerequisites, Key Construction Decisions

2. Creating High quality code Analysis [30%]
Design in Construction, Working Classes, High-Quality Routines, Defensive Programming, The Pseudo code Programming Process

3. Variables [10%]
General Issues in Using Variables, The power of Variable Names, Fundamental Data Types, Unusual Data Types

4. Statements [15%]
Organizing Straight-Line Code, Using Conditionals, Controlling Loops, Unusual Control Structures, Table-Driven Methods, General Control Issues

5. Code Improvements [20%]
The Software-Quality Landscape, Collaborative Construction, Developer Testing, Debugging, Refactoring, Code-Tuning Strategies, Code-Tuning Techniques

6. System Considerations [5%]
How Program Size Affects Construction, Managing Construction, Integration, Programming Tools

7. Software Craftsmanship [10%]
Layout and Style, Self-Documenting Code, Personal Character, Themes in Software Craftsmanship, Where to Find More Information

Main Reference Book(s):

1) Steve McConnell "CODE COMPLETE" Microsoft Press, Published by Wood Packer Publishers.

Suggested Additional Reading:

1) Bertrand Meyer, "Object Oriented Software Construction", Second Edition, PHI.

Chapter-wise coverage from main reference book:

Chapters: 1 – 35

Accomplishments of the student after completing the course:

At the end of the course the student will

- Learn and understand powerful software development techniques
- Learn the ways to keep big projects under control.
- Become Capable of maintaining and modifying their software and products successfully.

***** ***** *****

Course Name: Computer Based Optimization Models

Course Code: MCA 222

Objectives: Optimum utilization of scarce resources is an absolute necessity for the modern day organization – be it private or public, government or non-government, manufacturing industry or service industry, to achieve its objective of fulfilling the expectations of its stake holders. There are standard mathematical models which can be applied in such situations. These models range from simple deterministic models to more complex simulation based models. There are numerous mathematical techniques available to solve such models. The graduates of MCA course will be playing a key role not only in model formulation but also in applying these techniques using computers.
Hence, the objectives of this course is

- To provide the basic understanding of model formulation.
- To expose the students to the various techniques of optimization.
- To teach how to construct and solve mathematical models.

Prerequisites:

Basic knowledge of Matrix Algebra and Theory of Probability

Contents:

1. **Introduction to Operations Research [10%]**
   Operations Research – A Quantitative Approach to Decision Making, The History of Operations Research, Definitions of Operations Research, Features of Operations Research Approach, Operations Research Approach to Problem Solving, Models and Modeling in Operations Research, Advantages of Model Building, Methods for Solving Operations Research Models, Methodology of Operations Research, Advantages of Operations Research Study, Opportunities and shortcomings of the Operations Research Approach, Features of Operations Research Solution, Applications of Operations Research, Operations Research Models in Practice, Computer Software for Operations Research

2. **Linear Programming [10%]**
   Introduction, Structure of Linear Programming Model, Advantages of Using Linear Programming, Limitations of Linear Programming, Application Areas of Linear Programming, General Mathematical Model of Linear Programming Problem, Guidelines on Linear Programming Model Formulation, Examples of LP Model Formulation

3. **Linear Programming: The Graphical Method [10%]**
   Introduction, Important definitions, Graphical Solution Methods of LP Problems, Special cases in Linear Programming

4. **Linear Programming: The Simplex Method & Duality [20%]**
   Introduction, Canonical and Standard form of an LP Problem, Slack and Surplus Variables, Simplex Algorithm (Maximization Case), Simplex Algorithm (Minimization Case) (Excluding Two - Phase), Some Complications and Their Resolution, Types of Linear Programming Solution, Formulation of Dual LPP, Standard Results on Duality, Managerial Significance of Duality & its Advantages

5. **Transportation Problem & Assignment Problem [10%]**
   Introduction, Mathematical Model of Transportation Problem & Assignment Problem, Solution Methods of Assignment Problem, The Transportation Algorithm, Methods for Finding Initial Solution , Test for Optimality, Variations in Transportation Problem, Maximizations Transportation Problem, Trans-Shipment Problem, Variations of the Assignment Problem,

6. **Activity Network Analysis(PERT and CPM) [10%]**
   Introduction, Basic Difference Between PERT and CPM, Phases of Project Management, PERT/CPM Network Components & President Relationship, Critical Path Analysis, Project Scheduling With Uncertain Activity Times

7. **Deterministic Inventory Control Models [10%]**
   Introduction, The Meaning of Inventory Control, Functional Role Carrying Inventory Analysis, Reasons for Carrying Inventory, Factors Involved in Inventory Problem, Inventory Model building, Single Item Inventory Control Models Without Shortages, Single Item Inventory Control Models With Shortages.

8. **Queuing Theory (Single Server) [10%]**
   Introduction, The Structure of a Queuing System, Performance Measures of a Queuing System, Probability Distributions in queuing Systems, Classification of Queuing Models, Single-Server Queuing Models

9. **Simulation [10%]**
   Introduction, Simulation Defined, Types of Simulation, Steps of Simulation Process, Advantages and Disadvantages of Simulation, Stochastic Simulation and Random Numbers, Simulation of Inventory Problems, Simulation of Queuing Problems, Simulation of Investment Problems, Simulation of Maintenance Problems, Simulation of PERT Problems, Role of Computers in Simulation, Applications of Simulation

Main Reference Book(s):

1) Sharma J K “Operations Research – Theory and Applications” 4th Edition, MacMillan Business Books.

Suggested Additional Reading:

1) N D Vohra , "Quantitative Techniques in Management" , 3rd Edition, TMH..
2) Taha-Natarajan, Balasubramanie, Tamilarasi, "Operations Research", 8th Edition, Pearson Edu.
3) Shah, Gor, Soni, " Operations Research" , PHI.

Chapter wise Coverage from the Main Reference Book (s):

Chapters 1-5, 9, 10, 13(up to 13.6), 14(up to 14.8), 16(up to 16.6), 19, 25(up to 25.3)

Accomplishments of the student after completing the Course:

After completion of this course the student would be able to recognize the applicability of appropriate Operations Research Model to a real life problem, formulate the proper model and contribute significantly in solving the problem.

***** ***** *****

Course Name: Fundamentals of Networking

Course Code: MCA 223

Objectives:

- To introduce the basics of Computer Networks
- To understand the functionality of each layer of OSI and TCP/IP models and interactions between them
- To gain basic insight of programming for network solutions.

Prerequisites:

C Programming, Linux OS, Core Java

Contents:

1. Introduction [8%]
Uses of Computer Networks, Network Hardware, Network Software, Reference Models.

2. The Physical Layer [17%]
The Theoretical Basis for Data Communication, Guided Transmission Media, Wireless Transmission, Communication Satellites, Public Switched Telephone Network, Mobile Telephone System, Cable Television

3. The Data Link Layer [10%]
Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Protocols, Sliding Window Protocols

4. The Medium Access Control Sub Layer [20%]
The Channel Allocation Problem, Multiple Access Protocols, Ethernet, Wireless LANS, Broadband Wireless, Bluetooth, Data Link Layer Switching

5. The Network Layer [15%]
Network Layer Design Issues, Routing Algorithms, Congestion Control Algorithms, Quality of Service, Network Layer in the Internet

6. The Transport Layer [10%]
The Transport service, Elements of Transport Protocols

7. The Application Layer [5%]
Domain Name System, Electronic Mail

8. Network Security [15%]
Cryptography, Symmetric-Key Algorithms, Public-Key Algorithms, Digital Signatures.

Main reference Book(s):

1) Andrew S. Tanenbaum, “Computer Networks”, Fourth Edition, Pearson Edu.

Suggested Additional Reading:

1) Behrouz A. Forouzan, “Data Communication and Networking” Fourth Edition, TMH.
2) Prakash C. Gupta, “Data Communications and Computer Networks”, PHI
3) William Stallings, “Data & Computer Communications” Eighth Edition, Pearson Edu.
4) Natalia & Victor Olifer, “Computer Networks”, Wiley Student Edition
5) Cay S Horstmann, Gary Cornell, “Core Java 2, Volume 2 – Advanced Features”, Eighth Edition, Pearson Edu.
6) W. Richard Stevens, “Advanced Programming in Unix Environment”, Second Edition, Pearson Ed.

Chapter wise Coverage from Main Reference Book (s):

Chapters	Articles
1	1.1, 1.2, 1.3.1, 1.3.2, 1.3.3, 1.4.1, 1.4.2, 1.4.3
2	2.1.3, 2.2, 2.3, 2.4, 2.5.3, 2.6, 2.7
3	3.1 to 3.4
4	4.1, 4.2.2, 4.2.4, 4.2.6, 4.3 to 4.5, 4.6.1, 4.6.2, 4.6.7, 4.7.5, 4.7.6
5	5.1, 5.2.1 to 5.2.5, 5.2.9, 5.2.10, 5.3, 5.4.5, 5.6.7, 5.6.8
6	6.1.1, 6.1.2, 6.2.1, 6.2.2, 6.2.3
7	7.1, 7.2.4, 7.2.5
8	8.1 to 8.4

Practicals:

1) Framing Techniques
2) Error Detection and Correction Techniques
3) All Elementary DataLink Protocols
4) Shortest Path Routing
5) Cryptography

For Cryptography Programs use java.security, java.security.interfaces, java.security.spec Packages. Other programs are to be developed in C under Linux environment using named pipes and bitwise operators.

Accomplishments of the student after completing the Course:

After completion of this course students will be able to:

- Understand and use the process of protocols and other techniques using Java & ‘C’ under Linux environment.
- Analyze and develop protocol/algorithm to solve real problems.
- Able to implement various protocols, Framing Techniques, Routing Algorithm, Error detection and correction techniques, cryptography algorithms.
- Understand working of each layer of OSI and TCP/IP Model.

***** ***** *****

Course Name: Structured and Object Oriented Analysis and Design

Course Code: MCA 224

Objectives:

System Analysis and Design is a practical field that relies on a core set of concepts and principles. The objective of this course is to teach the students tried-and-tested techniques widely embraced by experienced analysts plus new and emerging tools and techniques that recent graduates are expected to apply on the job. The course is meant to give balanced exposure to both traditional and object oriented approaches to system analysis and design.

Prerequisite:

Fundamentals of Structured Programming and Fundamentals of Object Oriented Programming

Contents:

1. System Analysis Fundamentals [10%]

Types of Systems, Role of the System Analyst, Systems Development Life Cycle, CASE Tools, Interviewing, Joint Application Development, Using Questionnaires

2. Analysis Modeling [15%]

Data Flow Approach, Developing Data Flow Diagrams, Logical and Physical Data Flow Diagrams, Data Dictionary, Creating Data Dictionary, Using Data Dictionary, Process Specifications, Structured English, Decision Tables, Decision Trees

3. System Design [25%]

Designing Effective Output, Output Design Objectives, Designing a Web Site, Form Design, Web Forms Design, Data Concepts, Normalization, Denormalization, Data Warehouses, Human-Computer Interaction, Types of Interfaces, Dialog Design, Designing Queries, Effective Coding, Effective Data Capture, Input Validation

4. Object Modeling Concepts [15%]

Introduction, Modeling as a design technique, Class Modeling-Object and Classes, Association, Generalization, aggregation, Abstract class, Multiple inheritance, Metadata, Reification, Constraints, Derived data, Packages, State Modeling- State, Transitions and Conditions, State Diagrams, Nested state diagrams , Nested States, Signal Generalization, Concurrency

5. Object Oriented Analysis and Design [15%]

Process overview- Development stages, Development life cycle, System conception, Domain Analysis, Application Analysis, System Design, Class Design, Process summary

6. Basic Structural Modeling [10%]
Classes, Relationships, Common Mechanisms, Diagrams, Class Diagrams

7. Behavioral Modeling [10%]
Interactions, Use Cases, Use Case Diagrams, Interaction Diagrams, Activity Diagrams

Main Reference Book(s):

1) Systems Analysis and Design by Kendall & Kendall, PHI Publication, 7th Edition.
2) Object-Oriented Modeling and Design with UML by Michael Blaha, James Rumbaugh, Pearson Education Publication, 2nd Edition.
3) The Unified Modeling Language - User Guide by Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education Publication.

Suggested Additional Reading:

1) Modern Systems Analysis and Design by Jeffrey A. Hoffer, Joey F. George, Joseph S. Valacich, Prabin K. Panigrahi, Pearson Education Publication, 4th Edition.
2) Object Oriented Analysis and Design Using UML by Mahesh P. Matha, PHI Publication
3) UML Distilled by Martin Fowler, Pearson Edition, 3rd Edition.
4) Object Oriented Systems Development Using the Modified Modeling Language by Ali Bahrami, Tata McGraw Hill Publication.
5) System Analysis and Design with UML Version 2.0 by Alan Dennis, Barbary Haley Wixom, David Tegarden, John Wiley India Publication, 2nd Edition.
6) Appling UML and Patterns by Craig Larman, Pearson Education, 2nd Edition.
7) An Introduction to Object-Oriented Analysis – Objects and UML in Plain English by David William Brown, John Wiley & Sons Publication, 2nd Edition

Chapter-wise coverage from main reference book:

Book	Chapters
1	1,4,7-9,11-15
2	1-6,10-16
3	1-8,15-19

Accomplishments of the student after completing the course:

After completion of the course the students would be well versed with

- The role of System Analyst
- Modern structured analysis approaches
- Key modeling concepts that apply to both the traditional structured approach and the newer object-oriented approach
- Unified Process and use of UML for Object-Oriented Analysis and Design

***** ***** *****

Course Name: Distributed Application Development

Course Code: MCA 225

Objectives:

The objective of the course is to

- Develop proficiency in the advanced features of the Java Programming Language like RMI, JDBC, JavaBeans and Internationalization features.
- Be able to develop web applications using the Servlet Technology.
- Be able to parse XML files using the Java APIs using SAX Parser and the DOM model both.
- Be able to understand and use the JUnit framework for testing, and get an Idea about the Test Driven Development process.
- Be able to use the Ant tool for building and deploying Java applications.

Prerequisites:

Knowledge of the Java programming language, use of database and the SQL

Contents:

1. **Java Networking [5%]**
   - Connecting to a server
   - Implementing Servers
   - Interruptible Sockets
   - Making URL Connections

2. **Distributed Objects [10%]**
   - The Roles of Client and Server
   - Remote Method Calls
   - RMI Programming Model
   - Parameter Passing in Remote Methods

3. **Database Programming [10%]**
   - The Design of JDBC
   - The Structured Query Language
   - JDBC Configuration
   - Executing SQL Statements
   - Query Execution
   - Scrollable and Updatable ResultSets
   - Row Sets
   - Metadata
   - Transactions
   - Connection Management in Web
   - Introduction to LDAP

4. **Internationalization and Localization [5%]**
   - Locales
   - Number Formats
   - Date and Time
   - Collation
   - Message Formatting
   - Text Files and Character Sets
   - Resource Bundles

5. XML [5%]

- Introducing XML
- Parsing an XML Document
- Validating XML Documents
- Locating information with XPath
- Using Namespaces
- Streaming Parser
- Generating XML Documents
- XSL Transformations

6. JavaBeans Components [10%]

- Why Beans?
- The Bean-Writing Process
- Using Beans to Build an Application
- Naming Patterns for Bean Properties and Events
- Bean Property Types
- BeanInfo Classes
- Property Editors

7. Java Web Application Components [45%]

- Understanding Web Applications
  - Understanding Web Components
  - Servlets
  - JSP Pages
  - Introducing the MVC Design Pattern
  - Understanding Containers
  - Packaging Web Applications
  - Web Application Structure
  - JAR Files
  - WAR Files
  - HTTP
  - GET Request Method
  - POST Request Method
  - GET and POST in HTML Form Processing
  - Other Request Methods
  - The HTTP Response
  - Using Deployment Descriptors
- Working with Servlets
  - Introducing Servlet 2.4
  - Introducing Servlets and the MVC Pattern
  - Introducing the javax.servlet Package
  - Introducing the Servlet Interface
  - Introducing the GenericServlet Class
  - Introducing HTTP and Servlets
  - Understanding the Request/Response Cycle
  - Input and Output Streams
  - Introducing Servlet/Container Communication
  - Introducing the ServletContext
  - Understanding the Deployment Descriptor
  - Introducing the ServletContext Lifecycle Classes
  - RequestDispatcher interface
  - Using Filters and the RequestDispatcher

- Using Advanced Servlet Features
  - Understanding the Stateless nature of HTTP
  - Why Track Client Identity and State?
  - How Do You Maintain Sessions?
  - Session Management Using the Servlet API
  - Working with Filters
  - What is a Filter
  - Understanding Filters
  - Configuring Filters
  - Using Filters
  - Chaining Filters

8. Frameworks for Building Testing and Deploying Java Applications [10%]

- Introducing Ant
- A First Ant Build
- Understanding Ant datatypes and properties
- Testing with JUnit

Main Reference Book(s):

1) Cay S Horstmann, Gary Cornell, “Core Java 2, Volume 2 – Advanced Features”, Eighth Edition, Pearson Education
2) Sue Spielman and Meeraj Kunnumpurath, “Pro J2EE 1.4”, Wiley Computer Publishing (2004).
3) Steve Loughran, Erik Hatcher, “ANT In Action”, Second Edition, Wiley-Dreamtech/Manning.

Suggested Additional Reading:

1) Alexander V Konstantinou, William Wright, Chad Darby and others, “Beginning Java Networking”, Shroff Publishers and Distributors, (2001).
2) Jim Farley, William Crawford, Justin Gehtland, Prakash Malani and John G Norman, “Java Enterprise In A Nutshell”, Third Edition, Shroff Publishers and Distributors
3) Joseph O'Neil, “JavaBeans Programming – from the Ground Up”, Fourth Reprint, Tata McGraw-Hill Publishing Company Limited. (2000)
4) Marty Hall, Larry Brown, “Core Servlets and JavaServer Pages, Volume 1 : Core Technologies”, Second Edition, Pearson Education
5) Marty Hall, Larry Brown, “Core Servlets and JavaServer Pages, Volume 2 : Advanced Technologies”, Second Edition, Pearson Education
6) Steve Holzner, “Ant The Definitive Guide”, Second Edition, Shroff Publishers and Distributors

Unit wise Coverage from Main Reference Book(s):

Book	Chapter
1	3
1	10
1	4
1	5
1	2
1	8
2	2, 6, 7
3	1 – 4

Accomplishments of the student after completing the Course:

At the end of the course the student will be able to

- Create Java applications using advanced features like RMI, JDBC, JavaBeans and Internationalization.
- Develop and deploy web applications using the Servlet Technology.
- Develop code to parse XML documents using SAX parser and also using the DOM.
- Create Ant builds for java applications and run JUnit tests.

***** ***** *****

Course Name: Artificial Intelligence

Course Code: MCA 226(1)

Objectives:

The objective of the course is to

- Introduce the necessary understanding of human intelligence and to explore the mechanisms that enables the intelligent thought and action.
- Understand and learn effective ways for representing knowledge, applying intelligent problem solving techniques & searching techniques.

Prerequisites:

None

Contents:

1. Artificial Intelligence: Its Roots and Scope [5%]

AI: Early history, Overview of AI Application Areas, Attitudes towards Intelligence, knowledge and human ethics, Turing Test.

2. Problem Solving by Searching [15%]

Problem Solving Agents, Searching for Solutions, Real World Problems, Constraint Satisfaction Search, Uninformed and Informed Search Strategies, Searching with Partial Information, Heuristic Functions, Local Search Algorithms and Optimization Problems, Online Search Agents and Unknown Environments.

3. First Order Predicate Logic [10%]

First Order Logic Representation, Syntax and Semantics, Extensions and Notational Variations, Using First Order Predicate Logic, Knowledge Engineering and Applications of Predicate Logic.

4. Knowledge Representation [10%]

Semantic Nets, Frames, Conceptual Dependency, Scripts

5. Game Playing [10%]

Games, Optimal Decisions in Games, Minimax method and its complexity, Perfect and imperfect decisions, Alpha beta pruning and its effectiveness.

6. Intelligent Agents [5%]

Agents and Environment, Concept of Rationality, Nature of Environments, Structure of Agents.

7. Natural Language Processing [15%]

Understanding Natural Language, Deconstructing language, Syntax, Syntax and ATN parsers.

8. Expert Systems and Uncertainty Management in Expert Systems [10%]
Introduction to knowledge, Rules as knowledge representation technique, components of expert system, Characteristics, Advantages and disadvantages, Forward and Backward Chaining, conflict resolution, Real time example of Rule-based expert system.

9. PROLOG [20%]
Programs in PROLOG

Main Reference Book(s):

1) Stuart Russell and Peter Norvig "Artificial Intelligence-A modern Approach", 2nd Edition, Pearson Education.
2) George F. Lugar, "Artificial Intelligence-Structures and Strategies for Complex Problem Solving" 5th Edition, Pearson Education.
3) Michael Negnevitsky "Artificial Intelligence-A guide to Intelligent Systems" 2nd Edition, Pearson Education.
4) Elaine Rich, Kevin Knight, Shivashankar B. Nair, "Artificial Intelligence", 3rd Edition, McGraw Hill.
5) For PROLOG: Carl Townsend, "Introduction to Turbo PROLOG", BPB Publications.

Suggested Additional Reading:

1) Joseph C. Giarratano, "Expert Systems-Principals and Programming", 4th Edition in C Pearson Education.
2) N.P. Padhy, "Artificial Intelligence and Intelligent Systems", Oxford University Press.
3) Ivan Bratko, "PROLOG: Programming for Artificial Intelligence", 3rd Edition, Pearson Education

Chapter-wise coverage from main reference book(s):

Book	Chapters
1	2-4, 6,8
2	1, 14(up to 14.3)
3	2,3
4	9,10
5	1-16

Accomplishments of the student after completing the course:

At the end of the course the student will be able to

- Store and represent the knowledge in various applications and use different AI searching techniques.
- Deal with poorly defined or inexact problems that do not respond to the algorithmic solutions.

***** ***** *****

Course Name: Compiler Construction

Course Code: MCA 226(2)

Objectives:

The objective of the course is to

- To make students understand the phases of the compilation process and be able to describe the purpose and implementation approach of each phase.
- Give students practical exposure to aspects of theoretical computer science including Languages, Grammars, and Machines.
- Exercise and reinforce prior programming knowledge with a non-trivial programming project to construct a compiler (Lexical analyzer, parser, intermediate code-generation)

Prerequisites:

Discrete Mathematical Structures, Fundamentals of programming, Theory of Computation, System Software

Contents:

1. Introduction [5%]

Language processors, The structure of a compiler, The evolution of programming languages, The science of building compiler, Applications of compiler technology, Programming language basics.

2. Lexical Analysis [15%]

The role of Lexical Analyzer, Input buffering, Regular expression, Finite Automata (Deterministic & Non-deterministic), RE to NFA and NFA to DFA , Specification and recognition of tokens, a language for specifying lexical analyzer, Optimization of DFA-based pattern matchers, Design of lexical analyzer, The lexical analyzer generator Lex .

3. Syntax Analysis [20%]

Role of the parser, Context-Free Grammars, Writing Grammars, , Top Down parsing , Recursive Descent Parsing , Predictive Parsing, Bottom-up parsing, Shift Reduce Parsing, Operator Precedent Parsing, LR Parsers, SLR Parser, Canonical LR Parser, LALR Parser, Ambiguous Grammars, Parser generator Yacc.

4. Semantic Analysis [10%]

Syntax Directed definitions, Evaluation orders for SDD’s – dependency graphs, S-attribute definitions, L-attribute definitions, Applications of Syntax Directed Translation, Syntax Directed Translation schemes, Implementing L-Attribute SDD’s.

5. Intermediate Code Generation [15%]

Variants of Syntax Trees, Three –Address Code, Types and Declarations, Translation of expressions, Type checking, Control flow, Backpatching, Switch Statements, Intermediate Code for Procedures.

6. Run Time Environments [10%]
Storage Organization, Stack Allocation of Space, Access to Nonlocal Data on Stack, Heap Management, Various algorithms for Garbage collection

7. Code Generation [15%]
Issues in the design of code generator, The target Language, Addresses in the target code, Basic Blocks and Flow Graphs, Optimization of Basic Block , A simple Code generator, Peephole Optimization , Register Allocation and Assignment, Instruction Selection by expression Rewriting. Optimal Code generation for expressions, Dynamic programming Code Generation, Dynamic programming code generation

8. Code Optimization [10%]
Introduction, Principal Sources of Optimization, Data Flow Analysis, Constant Propogation, Partial-Redundancy Elimination , Loops in Flow Graph, Region-Based Analysis, Symbolic Analysis

Main Reference Book(s):

1) Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman, “Compilers - Principles, Techniques and Tools”, Second Edition , Pearson Education.

Suggested Additional Reading:

1) Kenneth C. Louden, “Compiler Construction: Principles and Practice”, Thompson Learning.
2) Charles N. Fisher, Richard J. LeBlanc Jr., “Crafting a compiler with C”, Pearson Education.
3) David Galles, “Modern Compiler Design”, Pearson Education.
4) Andrew W Apple “Modern Compiler Implementation in C”, Cambridge University Press, 1997.
5) Trembly, Sorenson, “Compiler Writing” , Tata Mc-Graw Hill Publication.

Chapter-wise coverage from main reference book:

Chapters: 1 – 9

Accomplishments of the student after completing the course :

At the end of the course the student will be able to

- Understand, design and implement a lexical analyzer.
- Understand, design and implement a parser.
- Understand, design code generation schemes.

- Understand optimization of codes and runtime environment.

***** ***** *****

Course Name: Parallel Processing

Course Code: 226(3)

Objectives:

- To develop proficiency in parallel methodologies
- To study and understand the technologies enabling parallel computing
- To study different parallel programming models

Prerequisites:

- Any programming language like C, C++
- Knowledge of Operating System

Contents:

1. **Introduction [15%]**
   Parallel processing, Parallelism in sequential machines, Parallel computers and architectures - Flynn's classification, UMA-NUMA architecture, pipelining, Vector processor, Programmability issues: Multi-processor OS, Overview of different programming models

2. **Methods of solving problems in parallel, Inter-task dependency [30%]**
   Temporal Parallelism, Data Parallelism (Static, dynamic, quasi-dynamic), Practical implementation using loop-splitting mechanism, self-scheduling mechanism, block-scheduling mechanism, Speedup expressions, Parallelism using specialist processors, Task graph, Task allocation to processors

3. **Need of synchronization [10%]**
   Contention problem, Race condition problem

4. **Dependency [10%]**
   Introduction to resource dependency, control dependency, Data dependency (true/flow, anti, output, input), Loop dependency (forward, backward, induction variable)

5. **Shard memory programming using processes and threads [10%]**
   Process/thread creation, Waiting for process/thread termination, Using shared memory, Using semaphores/mutex and condition variables under threads, Implementation of barrier

6. **Message passing Programming [15%]**
   Introduction, MPI and PVM: Introduction, Process creation, functionality of basic primitives, understanding of peer to peer and group communication

7. **Data parallel programming Language [5%]**
   Introduction to HPF

8. **Performance Tuning [5%]**
   Effect of Number of processes & Number of jobs (problem size), Effect of Placement of process-creation functions, Effect of use of cache memory Multi-core Technology

Practicals :
Practical implementation of parallel algorithms should be using processes and threads (POSIX standards) in ‘C’ under Linux environment.
Knowledge of UNIX system calls related to Creation of process, Creation of shared memory, Creation of semaphores, POSIX threads is required.

Main Reference Book(s):

1) M. Sashikumar, Dinesh Shikhare, P. Ravi Prakash, “Introduction to Parallel Procesing”, PHI
2) V. Rajaraman, C. Siva Rama Murthy, “Paralllel Computers Architechture and Programming”, PHI
3) Steven Brawer, “Introduction to Parallel Programming”, Academic Press

Suggested Additional Reading:

1) Meeta Gandhi, Tilak Shetty, Rajiv Shah, “Vijay Mukhi’s The ‘C’ Odyseey UNIX – The Open, Boundless C”, BPB
2) Cameron Hughes, Tracey Hughes, “Parallel and Distributed Programming using C++”, Pearson Education
3) Barry Wilkinson, Michel Allen, “Parallel Programming, Second edition, Pearson Education
4) W. Richard Steven, ”Advanced Programming in the UNIX Environment”, Pearson Education

Chapter wise Coverage from Main Reference Book (s):

Book	Chapters
1	1-6, 7, 9
2	1, 2 , 4 (upto 4.8.3), 6 (6.3 onwards)
3	4-8 (Upto 8.6), 10 (10.1 and 10.2), 11, Appendix C

Accomplishments of the student after completing the Course:

- Proficiency in parallel methodologies
- Optimize the speedup by applying parallel methodologies
- Understanding the technologies enabling parallel computing
- Learn different parallel programming models

***** ***** *****

Course Name: Computer Graphics

Course Code: MCA 226(4)

Objectives:

The objective of the course is to enable the students to

- Gain basic understanding of the underlying theory of graphics display.
- Develop basic algorithms for the applications of Computer Graphics.
- Explore and apply graphics algorithms and primitives using the OpenGL language.

Prerequisites:

- Fundamentals of programming.
- Knowledge of Co-ordinate Geometry and Matrix Algebra.

Contents:

1. A Survey of Computer Graphics [5%]

Graphs and Charts, Computer-Aided Design, Virtual Relative Environments, Data Visualizations, Education And Training, Computer Art, Entertainment, Image Processing, Graphical User Interfaces

2. Overview of Graphics Systems [5%]

Video-Display Devices, Raster Scan Systems, Graphics Workstations And Viewing Systems, Input Devices, Graphics Networks, Graphics On The Internet, Graphics Software, Introduction To OpenGL.

3. Graphics Output Primitives [20%]

Co-Ordinate Reference Frames, Specifying 2-D World Co-Ordinate Reference Frame In OpenGL, OpenGL Point Functions, OpenGL Line Functions, Line Drawing Algorithms, Parallel Line Algorithms, Setting Frame buffer Values, OpenGL Curve Functions, Circle Generating Algorithms, Fill Area Primitives, Polygon Fill Areas, OpenGL Polygon Fill Area Functions, OpenGL Vertex Arrays.

4. Attributes of Graphics Primitives [15%]

OpenGL State Variables, Color And Gray Scale, Point Attributes, Line Attributes, OpenGL Point Attribute Functions, OpenGL Line Attribute Functions, General Scan line Polygon Fill Algorithm, Scan Fill Of Convex Polygons And Regions With Curved Boundaries, Fill Methods For Areas With Irregular Boundaries, OpenGL Area With Fill Attribute Functions, Anti aliasing.

5. Geometric Transformations [20%]

Basic 2-D Geometric Transformation, Matrix Representations And Homogeneous Co-Ordinates, Inverse Transformations, 2-D Composite Transformation, Other 2-D Transformation, Geometric Transformation In 3-D Space, 3-D Translation, Rotation And Scaling, Composite 3-D Transformations And Other (Reflection And Shear) 3-D Transformation.

6. 2-D Viewing [15%]
The 2-D Viewing Pipeline, The Clipping Window, Normalization And View Port Transformation, Clipping Algorithm, 2-D Point Clipping, Line Clipping And Polygon Fill Area Clipping.

7. 3-D Viewing [10%]
Overview Of 3-D Viewing Concepts, The 3-D Viewing Pipeline, 3- Viewing Co-Ordinate Parameters, Transformation For World To Viewing Co-Ordinates, Projections Transformations, Orthogonal Projections, Oblique Parallel Projections, Perspective Projections.

8. 3-D Object Representations [10%]
Spline Representations, Cubic Spline Interpolation Methods, Bezier Spline Curves.

Main Reference Book(s):

1) Donald Hearn and M. Pauline Baker "Computer Graphics With OpenGL", Third Edition, Pearson Education.
2) Dave Shreiner, Woo, Neider and Tom Davis: "OpenGL Programming Guide – The Official Guide to Learning OpenGL, Version 2.1 "Sixth Edition, Pearson Education .
3) Dave Shreiner, Woo, Neider and Tom Davis: "OpenGL Programming Guide – The Official Guide to Learning OpenGL, Version 3.0 and 3.1" Seventh Edition, Pearson Education (to be announced).

Suggested Additional Reading:

1) James D Foley, Dam, Feiner And John Hughes " Computer Graphics – Principles And Practice", Second Edition in C Pearson Education.
2) Anirban Mukhopadhyay, Arup Chattopadhyay, "Computer Graphics and Multimedia",2nd Edition,Vikas Publishing House.
3) "Computer Graphics", ISRD Group. TMH.
4) Apurva A. Desai, "Computer Graphics", PHI.
5) F. S. Hill "Computer Graphics Using OpenGL", Third Edition PHI.

Chapter-wise coverage from main reference book:

Book	Chapters	Articles
1	1	Complete
1	2	2-1 to 2-9(Excluding 2-5)
1	3	3-1 to 3-9, 3-14 to 3-17
1	4	4-1 to 4-5, 4-7,4-8,4-10,4-11 to 4-14,4-17,4-18
1	5	5-1 to 5-5, 5-9 to 5-14
1	6	6-1 to 6-8 (Excluding 6-4)
1	7	7-1 to 7-8
1	8	8-8 (Excluding Spline Surfaces and Trimming Spline Surfaces), 8-9 (Exculding Cardinal Splines, Kochanek-Bartels Splines).
2	--	For help in OpenGL implementation.

Accomplishments of the student after completing the course:

At the end of the course the student will be able to

- Describe and explain different algorithms for Line Drawing, Circle Drawing, Line Clipping, Polygon Clipping and Polygon Filling.
- Use Matrices and Homogeneous Coordinates to represent and perform 2D and 3D transformations; Understand and use 3D To 2D Projection
- Understand Bezier Curves, Spline Interpolation
- Implement algorithms using OpenGL.

******* ***** *******

Course Name: Software Project Management

Course Code: MCA 226(5)

Objectives:

The course is aimed to help the students understand and appreciate the importance of Software Project Management in the software development life cycle and understand how to manage a software project.

- To understand the concept, process and importance of software project management.
- To gain knowledge of some of the CASE tools useful in project management.

Prerequisites:

None

Contents:

1. **Introduction to Project Management [5%]**
   Process based approach for Project Execution, Capability Maturity Model for Software

2. **Project Initiation [10%]**
   Proposals and Contracts, Requirement Specification and Management

3. **Project Planning [35%]**
   Process Definition and Tailoring, Process Database and Process Capability Baseline, Effort Estimation and Scheduling, Quality Planning and Defect Estimation, Risk Management, Project Management Plan, Configuration Management

4. **Project Execution and Termination [25%]**
   Life Cycle Execution, Peer Review, Project Monitoring and Control, Project Audits, Project Closure, ISO 9000, CMM, Software Process Improvement Project Management

5. **Introduction to CASE Tools [25%]**
   Introduction to MS Project, Visual Source Safe, MS Visio (UML Diagrams) and Mercury Suite of Testing Tools

Main Reference Book(s):

1) Pankaj Jalote , “CMM in Practice “, Pearson Education.
2) Teresa Stover, “MS Office Project 2007 Inside Out”, PHI.
3) Mark Walker, “MS Office Visio 2007 Inside Out”, PHI.

Suggested Additional Reading:

1) Roger Pressman, “Software Engineering – A Practitioner’s Approach”, 7th Edition , TMH.
2) Walker Royce, “Software Project Management A Unified Framework”, Pearson Education, First Impression.
3) Kathy Schwalbe, “Project Management in IT”, Cengage Learning, 2007. Second Indian Reprint 2009

Chapter wise Coverage from Main Reference Book (s):

Books	Chapters
1	1 -15, Appendix A – B
2	For introduction to MS Project 2007
3	For introduction to MS Visio 2007 (UML Diagrams)

Practicals:

Exercises in MS Project: Use Case, Sequence, Activity and Class diagrams in MS Visio.

Accomplishments of the student after completing the Course:

Upon completion of this course, the student will be able to:

- Plan and execute a software project.
- Use some of the important CASE tools useful in software project management.

***** ***** *****

Course Name: Data Warehousing & Data Mining

Course Code: MCA-226 (6)

Objectives:

The objective of the course is

- To understand the need of Data Warehouses over Databases, and the difference between usage of operational and historical data repositories.
- To be able to differentiate between RDBMS schemas & Data Warehouse Schemas.
- To understand the concept of Analytical Processing (OLAP) and its similarities & differences with respect to Transaction Processing (OLTP).
- To conceptualize the architecture of a Data Warehouse and the need for pre-processing.
- To understand the need for Data Mining and advantages to the business world. The validating criteria for an outcome to be categorized as Data Mining result will be understood.
- To get a clear idea of various classes of Data Mining techniques, their need, scenarios (situations) and scope of their applicability.
- To learn the algorithms used for various type of Data Mining problems.

Prerequisites:

Knowledge of RDBMS and OLTP

Contents:

1. **Introduction to Data Warehousing, A Multi-dimensional Data Model & Schemas [10%]**
   An overview and definition along with clear understanding of the four key-words appearing in the definition., Differences between Operational Database Systems and Data Warehouses; Difference between OLTP & OLAP, Overview of Multi-dimensional Model, and the basic differentiation between "Fact" and "Dimension", Star, Snowflakes, and Fact Constellations Schemas for Multi-dimensional Databases, Concept Hierarchies of "Dimensions", Parameters: Examples and the advantages, Concept of Multi-dimensional Cube, Pre-computation of Cubes, Constraint on Storage Space, Possible Solutions

2. **Operations in Multi-dimensional Models [8%]**
   OLAP Operations in Multi-dimensional Data Model: Roll-up, Drill-down, Slice & Dice, Pivot (Rotate), Type of OLAP Servers: ROLAP versus MOLAP versus HOLAP, Indexing OLAP Data: Bit-map Indexing, Concept of Metadata, Its uses in Selection of Appropriate Cube, etc.

3. **Data Warehouse Architecture [7%]**
   The Design of A Data Warehouse: A Business Analysis Framework, The Process of Data Warehouse Design, A 3-Tier Data Warehouse Architecture, Three Data Warehouse Models: Enterprise Warehouse, Data Mart, Virtual Warehouse

4. **Data Warehouse Implementation, Hypothesis-driven versus Discovery-driven Exploration of Data Cubes [5%]**
   Efficient Computation of Data Cubes, Indexing OLAP Data, Efficient Processing of OLAP Queries, Examples

5. **Pre-processing [8%]**
   The need for Pre-processing, Various Pre-processing Operations , Data Cleaning: Missing Values, Noisy Data, Inconsistent Data, Data Integration & Transformation

6. **Data Reduction Techniques [7%]**
   Data Cube Aggregation, Dimensionality Reduction, Data Compression: Basic Concepts only, Numerosity Reduction: Regression & Log-linear Models, Histograms, Clustering, Sampling

7. **Discretization and Concept Hierarchy Generation [5%]**
   For Numerical Data: Binning, Histogram Analysis, Cluster Analysis, Segmentation by Natural Partitioning, For Categorical Data

8. **Data Mining: Introduction [5%]**
   An Overview, Data Mining Primitives, A Data Mining Query Language - Examples

9. **Concept Description: Characterization & Comparison [8%]**
   Overview of Concept Description, Data Generalization & Summarization-based Characterization: Attribute-oriented Induction, Presentation of Derived Generalization, Analytical Characterization: An Example, Mining Class Comparisons: Discrimination between Different Classes: Class Comparison Methods & Implementations, Presentation of Class Comparison Descriptions, Class Descriptions: Presentation of both Characterization & Comparison

10. **Mining Association Rules in Large Databases [8%]**
    Association Rule Mining: An Overview & An Example, Basic Concepts, Apriori Algorithm, From Association Mining to Correlation Analysis

11. **Classification & Prediction [15%]**
    An Overview, Basics of Supervised & Unsupervised Learning, Difference between Classification & Prediction, Criteria for Comparing Classification Methods, Classification by Decision Tree Induction, Tree Pruning, Extracting Classification Rules from Decision Trees, Bayesian Classification: Bayes Theorem, Naïve Bayesian Classification, Bayesian Belief Networks, An Overview of Other Classification Methods, Prediction: Linear & Multiple Regression, Non-linear Regression, Other Regression Models, Classifier Accuracy: Estimating the Accuracy, Increasing the Accuracy, Accuracy Parameters – Sensitivity, Specificity, Precision

12. Cluster Analysis [6%]

An Overview, Typical Requirement of Clustering in Data Mining, Types of Data in Cluster Analysis: Interval-scaled Variables, Binary Variables, Nominal, Ordinal & Ratio-scaled variables, Variables of Mixed Types, Categorization of Major Clustering Methods

13. Outlier Analysis [5%]

An Overview, Statistical-based Outlier Detection, Overview of Distance-based Outlier Detection & Derivation-based Outlier Detection

14. Data Mining System Products & Research Prototypes [3%]

How to Choose a Data Mining System, Examples of Commercial Data Mining Systems

Main Reference Book (s)

1) Jiawei Han & Micheline Kamber, "Data Mining: Concepts & Techniques", Second Edition, Morgan Kaufmann Publishers

Suggested Additional Reading

1) Paulraj Ponniah, "Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals", Wiley-India
2) Daniel T. Larose, "Data Mining Methods & Models", Wiley-India
3) Vikram Pudi & P. Radhakrishnan, "Data Mining", Oxford University Press
4) Alex Berson & Stephen J. Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw-Hill
5) Michael J. A. Berry & Gordon S. Linoff, "Data Mining Techniques", Wiley-India
6) Pang-Ning Tan, Michael Steinbach, Vipin Kumar, " Introduction to Data Mining", Pearson Education
7) A B M Shawkat Ali, Saleh A. Wasimi, " Data Mining : Methods and Techniques", Cengage Learning
8) Rajan Chattamvelli, "Data Mining Methods", Narosa Publishing House
9) Sam Anahory, Dennis Murray , "Data Wharehousing in the real World", Pearson Education
10) Pieter Adriaans, Dolf Zantinge, "Data Mining", Pearson Education
11) G.K. Gupta , " Introduction to Data Mining with Case Studies", PHI

Chapter and Section-Wise Coverage from Main Reference Book:

Chapters	Articles
1	1.7 (incl. Fig 1.13)
2	2.1, 2.3.1, 2.3.2, 2.4,2.5.1, 2.5.3, 2.5.4, 2.6
3	3.1, 3.2, 3.3, 3.4
4	4.3
5	5.1, 5.2.1, 5.2.2, 5.4
6	6.1, 6.2, 6.3.1,6.3.3, 6.4.1, 6.4.2, 6.4.3, 6.5.2, 6.11, 6.12, 6.13, 6.14
7	7.1, 7.2, 7.3, 7.11
11	11.2

Accomplishment of Students after Completing the Course

- Ability to create a Star Schema for a given Data Warehousing requirements
- Ability to decide the number & levels of pre-computed Data Cubes, the corresponding Metadata, and the appropriate OLAP operation
- Ability to apply pre-processing on existing operational & historical data for creation of Data Warehouse
- Ability to apply Apriori algorithm for Association Mining
- Ability to apply Decision Tree and Bayesian algorithms for Classification
- Ability to mine Statistical Measures in large databases
- Ability to differentiate between Classification & Clustering, and similarly between Supervised Learning & Unsupervised Learning

***** ***** *****