MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
(M.Sc.IT)

PROGRAMME GUIDE

INDEX

INTRODUCTION

The Master of Science in Information Technology (M.Sc.IT) is a Programme designed to meet the needs of the market for expertise in Information Technology (IT). The Programme is intended to address the increasing demand in the work-place for IT professionals with a broad and sound knowledge of both technical and managerial skills. A **master's degree** is an granted to individuals who have undergone study demonstrating a mastery or high-order overview of a specific area.

ACADEMIC OBJECTIVES

- To equip postgraduate students with an integrated set of skills that will allow them to develop their professional careers in Information Technology.
- To equip students with the theoretical and practical knowledge that is necessary to enable them to understand the design of complex computer applications/science.
- The program also prepares students to embrace future developments in the field and has a demonstrated professional relevance.
- The program helps students to acquire the latest skills and build their future capabilities using world-class technology. At the end of this program, the student will possess a strong foundation of computer systems and information technology.
- Dexterity in advanced programming languages; power to build sophisticated software for wide area of applications.
- Skills to work with higher end applications in internet technologies; also managerial ability to analyze, design, develop and to maintain software development.

PROGRAMME CODE: 1423

DURATION OF THE PROGRAMME:

Minimum Duration   **2** years
Maximum Duration   **5** years

MEDIUM OF INSTRUCTION/ EXAMINATION:
Medium of instruction and Examination shall be **English**.

COURSE CODE	COURSE TITLE	Cr.	CA	ETE(Th.)	ETE(Pr.)
M.Sc.(IT) (Master of Science in Information Technology) Scheme					
TERM 1					
DCAP401	FOUNDATIONS OF COMPUTER PROGRAMMING	4	20	60	20
DENG401	ADVANCED COMMUNICATION SKILLS	4	20	80	0
DMGT409	BASIC FINANCIAL MANAGEMENT	4	20	80	0
DCAP402	DATABASE MANAGEMENT SYSTEMS	4	20	60	20
DCAP403	OPERATING SYSTEM	4	20	80	0
TERM 2					
DCAP404	OBJECT ORIENTED PROGRAMMING	4	20	60	20
DCAP405	SOFTWARE ENGINEERING	4	20	60	20
DCAP406	COMPUTER NETWORKS	4	20	80	0
DCAP409	ANALYSIS & DESIGN OF INFORMATION SYSTEMS	4	20	80	0
DCAP408	WEB PROGRAMMING	4	20	60	20
TERM 3					
DCAP501	MODERN PROGRAMMIMG TOOLS & TECHNIQUES-I	4	20	60	20
DCAP407	DATA STRUCTURE	4	20	60	20
DCAP502	COMPUTER ORGANIZATION AND ARCHITECTURE	4	20	80	0
DCAP511	E-COMMERCE & E-BUSINESS	4	20	80	0
DCAP504	COMPUTER GRAPHICS	4	20	60	20
TERM 4					
DCAP505	MODERN PROGRAMMING TOOLS & TECHNIQUES-II	4	20	60	20
DCAP506	ARTIFICIAL INTELLIGENCE	4	20	80	0
DCAP507	SYSTEM SOFTWARE	4	20	80	0
DCAP512	WAP & WML	4	20	60	20
DCAP606	BUSINESS INTELLIGENCE	4	20	80	0
TOTAL CREDITS		**80**			

Course Code:	**D**	**C**	**A**	**P**	**4**	**0**	**1**	Course Title:	**FOUNDATIONS OF COMPUTER PROGRAMMING**

	WEIGHTAGE		
	CA	**ETE (Pr.)**	**ETE (Th.)**
	20	**20**	**60**

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction:** ANSI C standard, Overview of Compiler and Interpreters, Structure of C Program ,Programming rules, Execution
2.	**Basics-The C Declarations:** C Character Set, keywords, **:** Identifiers, data types, operators, constants and variables **Operators & Expressions**
3.	**Input/ Output in C:** Formatting input & output functions.
4.	**Decision making statements** – if, else if **Control Statements:** For, do while, while. Control transfer statements - break, continue.
5.	**Arrays and Strings**: Defining arrays; I/O of arrays, I/O of string data; built-in library functions to manipulate strings, array of strings
6.	**Pointer**: Introductions, Features, Declaration, Pointers and Arrays, pointers to pointers ,Pointers and strings, Void Pointers
7.	**Functions:** Defining and accessing a functions, passing arguments – call by value, function prototypes, recursive functions **Storage Classes:** Storage classes and their usage
8.	**Structures & Unions:** Defining and processing structures, array of structures, nested structures, Unions & difference from Structures
9.	**Files:** Opening, reading, writing & Closing file
10.	**Additional In C:** Dynamic memory allocation, Memory models, Linked List

LABORATORY WORK:

1. Implementation of C Programming Concepts (Operators, Data types, Control Statements, Functions, Arrays, Strings, Structures, Union, Pointers, File Handling)

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Ashok N. Kamthane, "Programming with ANSI & Turbo C", Pearson Education, Year of Publication: 2008
2. Byron Gottfried , "Programming With C", Tata McGraw Hill Publishing Company Limited, New Delhi
3. B.W. Kernighan and D.M. Ritchie, "The C Programming Language", Prentice Hall of India, New Delhi
4. E.Balagurusamy , "Programming in ANSI C ", Tata McGraw Hill Publishing Company Limited, New Delhi.
5. Behrauz A.Foruzan & Richard F.Gilberg , " Computer science – A structure programming approach Using C ", Thomson Asia , 2001.

Course Code:	D	E	N	G	4	0	1	Course Title:	**ADVANCED COMMUNICATION SKILLS**

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
A	**Speaking Skills** ---to enhance the basic speaking skills, one needs apt language and the correct pronunciation. • Simple rules of pronunciation and intonation • Formal oral presentations--- Power point presentations or presentations using other visual aids followed by actual practice of it. **Interview Skills**—[Types of interviews , employer's expectations, types of questions, some standard questions , answering techniques, mock interviews]. **Working with Customers**--- essential speaking elements needed to communicate with the customers—[apt questions to determine the context, apt responses to put them at ease, apt responses to acknowledge their efforts, using listener cantered language , asking questions to understand their problems , establish rapport , denying requests, coping with angry customers. **Improving Informal Communication**—speaking persuasively, negotiating effectively, managing conflicts. Formal and regularly used expressions in given situations. **Group Discussions**---- a detailed briefing of do' s and don'ts followed by GD's based on topics relevant to their field. Kinds of GD's—to convey information or to instruct or solve problems or to take decisions
B	**Reading Skills—**skills we need to read successfully **Reading Strategies / Techniques / Types:** equipped with separate and adequate reading passages to practice the skill **Comprehension of Written Texts :** selecting information, identifying topic –shift, cause – effect, point of view [the texts are articles / editorials etc., from varied streams of subjects] **Aesthetic Reading Skill-** poem-- "Raisin In the Sun" – Langston Hughes --"ways to kill a Man"—Edwin Brock
C	**Writing Skills** – to reinforce the grammatical structures Grammar – Subject – Verb agreement, Basic sentence patterns, Conditional sentences, Subordinating Conjunctions, Correlative Conjunctions, One Word Substitutes
D	**Writing skills --**to enhance formally structured effective official writing • Understanding Reports and Proposals • Types of reports • Structure and Layout of a Formal Report—writing the beginning / the body / end matter • Business Reports • Writing Short Reports • Proposal Writing and Process Description • Technical Proposals • Writing Proposals • Supplementary Parts / Appended Parts • Citing sources

READINGS: SELF LEARNING MATERIAL.

Course Code:	D	M	G	T	4	0	9	Course Title:	**BASIC FINANCIAL MANAGEMENT**

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
1.	Meaning, Objectives and Scope of Financial Management
2.	**Finance Functions:** Investment, Financing, Liquidity & Dividend Decisions, Risk & Return Trade Off.
3.	**Sources of Finance:** Long term, Medium term & short term;Time Value of Money: Basic Concepts
4.	**Cost of Capital:** Concept and its significance, measurement of cost of capital of various sources of funds. Weighted average cost of capital.
5.	**Capital Structure Decision:** Understanding debt and equity.
6.	Theories of Capital Structure, Optimum Capital Structure.
7.	**Capital Budgeting:** Analytical study of various methods of Capital Budgeting.
8.	**Working Capital:** Concept and Significance, Determining working capital requirements; Basics of receivables, Inventory and Cash Management.
9.	**Dividend Policy:** Determinants of Dividend Policy, Theories of dividend and Forms of dividend.
10.	Break Even Analysis.

READINGS: SELF LEARNING MATERIAL

ADDITIONAL READINGS:

1. Shrivastava Rajiv and Mishra Anil, Financial management, Oxford Publications, 2009.
2. Sharan Vyupkesh, Fundamentals of Financial Management, Pearson Education, 2009.
3. Reddy G. Sudarshana, Financial Management, Principles and Practice, 2008.
4. Chandra, Prasana, Financial Management, Tata McGraw Hill, 7e.

Course Code:	D	C	A	P	4	0	2	Course Title:	DATABASE MANAGEMENT SYSTEMS

	WEIGHTAGE		
	CA	ETE(Pr.)	ETE (Th.)
	20	20	60

COURSE CONTENTS:

Sr. No.	Topics
1.	**Database Fundamentals:** Database systems, Database Architecture **Relational Model**, Structure of Relational databases, fundamental, additional and extended relational algebra operations
2.	**SQL:** Data Definition, datatypes, schema definition, Basic structure of SQL Queries, Creating tables, DML operations, DDL commands for creating and altering, Set Operations, Aggregate Functions, NULL values
3.	**Advanced SQL:** Subqueries, Nested subqueries, Complex queries, Views, Joined relations, Integrity constraints, Authorization, DCL Commands, Embedded SQL, Dynamic SQL
4.	**Relational Languages:** Tuple Relational calculus, Domain relational calculus, Query by Example **Database design and ER model:** Overview of Design process, Entity relationship model, constraints, ER Diagrams, ER Design issues, Weak entity sets, extended ER features
5.	**Relational Database Design:** Features, Atomic Domains and first normal form, Functional dependency theory decomposition using functional dependencies, decomposition using Multivalued dependencies, database design process **Normalization:** 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.
6.	**Transaction Management:** Concept of Transaction, Transaction State, Implementation of atomicity and durability, concurrent execution, Serializability, Recoverability, Implementation of Isolation, testing for Serializability. **Concurrency Control**: Lock based protocols, Timestamp based protocols, Validation based protocols, Deadlock handling, Insert and Delete operations, Weak levels of consistency
7.	**Recovery system:** Failure classification, storage structure, recovery and atomicity, log-based recovery, recovery with concurrent transactions, buffer management, failure with loss of non-volatile storage
8.	**Query Processing:** Overview, measures of query cost, selection operation, sorting, join operation, evaluation of expressions **Query Optimization:** Transformation of relational expressions, estimating statistics of expression results, Choice of evaluation plans
9.	**Parallel Databases:** I/O parallelism, Interquery parallelism, Intraquery parallelism, Interoperation parallelism, Interoperation parallelism
10.	**Application development and administration**: web interfaces to databases, performance tuning

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Author: Silberschatz–Korth–Sudarshan: Database System Concepts, Fourth Edition, Title: Database System Concepts, Publishers: Tata McGraw Hill.
2. Elmasri & Navathe, Fundamentals of Database systems, Addison & Weisely, New Delhi.
3. C. J. Date, Database Systems, Prentice Hall of India, New Delhi.
4. Martin Gruber, Understanding SQL, BPB Publication, New Delhi.
5. Val Occardi, Relational Database: Theory & Practice, BPB Publication, New Delhi.
6. Ivan Bayross, SQL, PL/SQL The Programming Language of Oracle, BPB Publication.

Course Code:	**D**	**C**	**A**	**P**	**4**	**0**	**3**	Course Title:	**OPERATING SYSTEM**

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction:** Operating system Meaning, Supervisor & User mode, operating system operations & Functions, Types of OS: Single-processor system, multiprogramming, Multiprocessing, Multitasking, Parallel, Distributed, RTOS etc.
2.	**Operating System Structure:** OS Services, System Calls, System Programs, OS Structures, layered structure Virtual machines,
3.	**Processes**: Process Concept, PCB, Operation on Processes, Cooperating Processes, Inter process Communication, Process Communication in Client Server Environment. **Threads:** Concept of Thread, Kernel level & User level threads, Multithreading, Thread Libraries, Threading Issues
4.	**Scheduling:** scheduling criteria, scheduling algorithms, Type of Scheduling: Long term, Short term & Medium term scheduling, multi-processor scheduling algorithm, thread scheduling,
5.	**Process Synchronization:** Critical Section problem, semaphores, monitors, Deadlock characterization, Handling of deadlocks -deadlock prevention, avoidance, detection, recovery from deadlock.
6.	**Memory Management**: Logical & Physical Address space, Swapping, Contiguous memory allocation, paging, segmentation, Virtual memory, demand paging, Page replacement & Page Allocation algorithms, thrashing, Performance issues
7.	**File Management:** File concepts, access methods, directory structure, file system mounting, file sharing, protection, Allocation methods, Free space Mgt., Directory Implementation.
8.	**I/O & Secondary Storage Structure:** I/O H/W, Application I/O Interface, Kernel I/O subsystem, Disk Scheduling, disk management, swap-space management, RAID structure.
9.	**System Protection:** Goals of protection, Access matrix and its implementation, Access control and revocation of access rights, capability-based systems
10.	**System Security:** Security problem, program threats, system and network threats, cryptography as a security tools, user authentication, implementing security defenses, firewalling to protect systems and networks. Case studies Windows OS, Linux or any other OS

READINGS: SELF LEARNING MATERIAL.
ADDITIONAL READINGS:

1. Silberschatz, Gagne & Galvin, "Operating System Concepts", John Wiley & Sons, Seventh Edition or Latest
2. A.S. Tanenbaum : Operating System : Design and Implementation, Prentice Hall of India.
3. Milankovic, Operating system, Tata Macgraw Hill, New Delhi.
4. Stalling, W., "Operating Systems", 2nd edition, Prentice Hall.
5. Deitel H. M., "Operating Systems, 2nd edition, Addison Wesley.

Course Code:	D	C	A	P	4	0	4	Course Title:	**OBJECT ORIENTED PROGRAMMING**

WEIGHTAGE		
CA	**ETE (Pr.)**	**ETE (Th.)**
20	**20**	**60**

COURSE CONTENTS:

Sr. No.	Topics
1.	**Review:** Review of basic concepts of object-oriented programming & Introduction of OOP Languages, Comparison between procedural programming paradigm and object-oriented programming paradigm.
2.	**Beginning with OOP Language:** Review of Tokens, Expressions, Operators & Control Structures. Scope Resolution operator, member dereferencing operator, Reference Variables Review of Functions, Function Overloading, Inline Functions, Default Arguments
3.	**Classes & Objects:** specifying a class, Defining member functions, creating class objects, accessing class Members. Access specifiers – public, private, and protected Classes, its members, objects and memory allocation
4.	Static members, the const keyword and classes, the static objects. Friend Function & its usage Empty classes, nested classes, local classes
5.	**Constructors & Destructors:** Need for constructors and destructors, copy constructor, dynamic constructors, Destructors, constructors and destructors with static members
6.	**Operator Overloading & Type Conversion:** Defining operator overloading, rules for overloading operators, Overloading of unary operators and various binary operators with friend functions and member functions Type conversion – basic type to class type, class type to basic type, class type to another class type
7.	**Inheritance:** Introduction, defining derived classes, forms of inheritance, Ambiguity in multiple and multipath inheritance, virtual base class, Overriding member functions, order of execution of constructors and destructors Virtual functions & Polymorphism: virtual functions, pure virtual functions, abstract classes, introduction to polymorphism
8.	**Pointers & Dynamic Memory Management:** understanding pointers, accessing address of a variable, declaring & initializing pointers, Pointer to a pointer, pointer to a function, dynamic memory management new and delete operators, this pointer
9.	**Console I/O:** concept of streams, hierarchy of console stream classes, Unformatted I/O Operations, Managing output with manipulators
10.	**Working with Files:** Opening, Reading, Writing, Appending, Processing & Closing difference type of files, Command line Arguments

LABORATORY WORK:

Sr. No.	Topics
1.	Implementation of Concepts of OOP using C++ covered in the syllabus

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Robert Lafore, "Object Oriented Programming with C++", Galgotia.
2. Author: Herbert Schildt,Title: Teach Yourself C++,Publishers: Tata Mc Graw Hill,Year of   Publication: 2005.
3. J Marget A. Ellis and Bjarne Stroustrup ,The Annotated C++ reference manual, Addison Wesley New York.
4. Waite Group Lafore R.,Object oriented programming in C++, Waite Group Lafore R.
5. Lippman F. B.C++ Primer, Addison Wesley
6. E. Balagurusamy, "Object Oriented Programming with C++", Tata McGraw Hill

Course Code:	**D**	**C**	**A**	**P**	**4**	**0**	**5**	Course Title:	**SOFTWARE ENGINEERING**

COURSE CONTENTS:

WEIGHTAGE		
CA	ETE (Pr.)	ETE (Th.)
20	20	60

Sr. No.	Topics
1.	**Introduction to Software Engineering:** The Evolving Role of Software, Software, Software Myths.
2.	**A generic view of Process:** Software Engineering-A Layered Technology, A process framework, The Capability Maturity Model Integration, Process Patterns, Process Assessment.
3.	**Process Models :** Prescriptive Models, The Waterfall model, Incremental Process Models : The Incremental model, The RAD model Evolutionary Process models: Prototyping, The Spiral model, The Concurrent Development model, A final comment on evolutionary Processes.
4.	**An Agile view of Process :** What is Agility, Agile Process models : XP, ASD, DSDM, Scrum, Crystal, FDD, AM. **Requirements Engineering:** A Brigade to design & construction, Requirements Engineering tasks: Inception, Elicitation, negotiation, Specification, Validation, Requirements Management .
5.	**Software Engineering Practice :** The Essence of practice, Core Principles, Planning practices, Modelling practices: Analysis modelling principles, Design Modelling principles Construction practice : Coding principles and concepts, Testing principles **Design Engineering:** Design process & Design Quality**. Design Concepts:** Abstraction ,Architecture, patterns, Modularity, Information hiding, Functional independence, Refinement, Refactoring, Design Classes
6.	**System Engineering :** The System Engineering Hierarchy: System Modelling, System Simulation, System Modelling: Hatley-Pirbhai Modelling, System Modelling with UML **Creating an Architectural Design:** Data design: Data design at the Architectural level & component level, Architectural Design: Representing the system in Context, Defining Archetypes, Refining the Architecture into components, Describing installations of the system.
7.	**Testing Strategies: -** Testing strategies for conventional software, test strategies for object-oriented software, validation testing, system testing. **Requirements Engineering:** A Bridge to Design and Construction, Requirements Engineering Tasks: Inception, Elicitation, Elaboration, Negotiation, Specification, Validation, Requirements Management.
8.	**Testing Tactics**: Black-box testing & white box testing, flow-graph testing, equivalence partitioning, Boundary value analysis, Fault based testing. **Building the Analysis Model:** Requirements Analysis: Overall objective and philosophy, Analysis rules of Thumb, Domain Analysis, Analysis Modelling approaches. Data Modelling concepts: Data objects, Data attributes, Relationships, Cardinality and Modality
9.	**Design Engineering :** Design Process and Design quality, Design concepts: Abstraction, Architecture, patterns, Modularity, Information hiding, Functional independence, Refinement, Refactoring, Design classes
10.	**Creating an Architectural Design :** Data design: Data design at the Architectural level and Component level, Architectural Design: Representing the system in Context, Defining Archetypes, Refining the Architecture into Components, Describing installations of the system

LABORATORY WORK:

Sr. No.	Topics
1.	Various Tools available for CASE (Computer Aided Software Engineering).
2.	Practical to show how to create an information gathering document.
3.	Finalizing the SRS Document based upon the information gathered and analysis of the same.
4.	How to Create Use Cases.
5.	How to Create ER Diagrams.
6.	Developing Test Cases and Test Plan.

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. R.S. Pressman, Title: Software Engineering – A Practitioner's Approach
   Publishers: McGraw Hill, 6th edition
2. P. Jalote, "An Integrated approach to Software Engineering", Narosa.
3. R. Fairley, "Software Engineering Concepts", Tata McGraw Hill, 1997.
4. Software Engineering by Ian Sommerville, Pearson Education.

Course Code:	D	C	A	P	4	0	6	Course Title:	**COMPUTER NETWORKS**

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction to Computer Networks:** uses of computer networks,
2.	Network hardware, network software, Reference models, Example networks
3.	**Physical Layer :** Theoretical Basis for Data Communication, Guided Transmission Media, Wireless Transmission, Communication Satellites
4.	Public Switched Telephone Network, The Mobile Telephone System, Cable television
5.	**Data Link Layer:** Design Issues, Error Detection and Correction
6.	Elementary data link protocols, Sliding – Window protocols, Protocol verification, Example Data Link Protocols
7.	**The Medium Access Control Sub Layer**: The Channel Allocation Problem
8.	Multiple Access Protocols, Ethernet, wireless LANs, Bluetooth, Data Link Layer Switching.
9.	**Network Layer:** Design Issues, Routing Algorithms, Internetworking, network Layer in the Internet, Congestion Control Algorithms, Quality of service
10.	**Transport Layer:** Transport Service, Elements of Transport Protocols, The internet transport protocols: UDP,TCP **Application Layer:** DNS ,E-mail, The World Wide Web, Multimedia ,**Network Security** - Cryptography

READINGS: SELF LEARNING MATERIAL.
ADDITIONAL READINGS:

1. A.S. Tananbaum, "Computer Networks", Pearson Education, Delhi, Fourth edition, Year of Publication: 2009.
2. Behnouz A. Forouzan, "Data Communication and networking", 2nd Ed. Update, Tata McGraw Hills 2003
2. Black U, "Computer Networks-Protocols, Standards and Interfaces", PHI 1996
3. Comer E. Doughlas, "Computer Networks and Internets", 2nd Ed., Pearson, 2000
4. W. Stallings, "Data and Computer Communications", 7th Ed., Pearson, 2002.
Laura Chappell (Ed), "Introduction to Cisco Router Configuration", Techmedia, 99

Course Code:	D	C	A	P	4	0	9	Course Title:	ANALYSIS & DESIGN OF INFORMATION SYSTEMS

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction to information systems development** –System Analyst, System Analysis & Design, Categories of Information Systems, System Development Strategies, Implementation and Evaluation
2.	Managing the application development portfolio-Information system Planning, Managing project review & Selection Information Systems & User-groups Committee Methods
3.	**Analysis:** Preliminary Investigation, Scope of Study, Conducting the investigation, Testing Project Feasibility, Handling infeasible projects
4.	**Tools for System Requirements**: Requirement Determination, Activities, Types. **Fact-finding techniques: Interview, Questionnaire, Record Review, Observation.** Tools for documenting Procedures and Decisions: Decision Trees, Decision Tables, Structured English
5.	**Structured Analysis Development Strategy:**, Features, Data Flow Tools.
6.	**Tools for Structured Design:** Data Flow Diagrams, Data Dictionaries.
7.	**Application Prototypes:** Purpose, Steps, Use, Tools. Prototype Example. **Computer Aided System Tools:**- Role, Categories, CASE Tools.
8.	**Analysis To Design transition**- Objectives, Features. Element of Design: Output, Files, Database Interaction, Input, Control, Procedures, Program Specifications.
9.	**Design of Computer output:** Objective, Needs, Types. Design Input and Control: Objectives Capturing input data, Input validation.
10.	Design of Online dialogues & its interface, design of files & Use of Auxiliary storage devices **Systems Engineering & Quality Assurance**:-Design Objectives, Design of Software & Documentation.

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Author: James A. Senn,Title: Analysis and Design of Information Systems, Publishers: Tata Mcgraw Hill,Year of Publication:2004
2. Whitten, Bentley and Barlow, "System Analysis and Design Methods", Tata Mcgraw Hill.
3. Robert J. Thierauf, "System Analysis and Design- A Case Study Approach", CBS.
4. James Rambaugh, Grady Booch, Jacobson, "The Unified Modelling Language Reference Manual", Object Tech Series.
5. Elias M Awadh, "System Analysis & Design", Galgotia Publication

Course Code:	D	C	A	P	4	0	8	Course Title:	**WEB PROGRAMMING**

WEIGHTAGE		
CA	ETE (Pr.)	ETE (Th.)
20	20	60

COURSE CONTENTS:

Sr. No.	Topics
1.	**Internet Fundamentals**: Introduction to Internet, Web browser, web page, website, homepage, hyperlinks, hypermedia, HTTP, WWW, Web server, Client server architecture model for web requests, URL
2.	**Creating static web pages:** HTML document structure, singular and paired tags, test formatting, hyperlinks, adding images, audio and video, creating lists, tables, forms, frames, using multiple windows for web pages
3.	**Cascading Style Sheets:** Style tag, DIV and SPAN, Internal and External stylesheets, Creating and using Classes, applying style on text and images
4.	**Scripting Language** Java Script programming, Data Types, Variables, Arrays, Operators. Loops, functions, Dialog boxes, String Manipulation functions, Using Timer in web page. Setting and Getting date object in a web page.
5.	**DOM Model**. Events handling through JavaScript, How to use forms in JavaScript
6.	**ASP:** introduction to asp, installing IIS, ASP variable, ASP operators, conditional, loops and case statements and arrays
7.	**ASP Web Forms:** Introduction to CGI, Client side and server side scripting, building and processing web forms
8.	**ASP Objects**: Response, Request, Server, Session, Application. Purpose of Global.asa file, #include, Recordset objects
9.	**ASP Cookies and Caching** Procedures, Cookies, ASP file system, send e-mail, **Caching**: page, data, fragment, output.
10.	**Database Connectivity:** Open and Close a connection, reading from the database, inserting, deleting and updating the database records Building Database Applications Using ActiveX Data Objects

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Teach Yourself ASP in 21 Days: Sams publishing.
2. Author: Bayros Ivan, Title: Web Enabled Commercial Application Development Using HTML, DHTML, JavaScript, Publishers: BPB publications.
3. Teach Yourself HTML 4 With XML, DHTML and Java Script - Stephine Cottrell Bryant.
4. An Introduction to Apache : Tata McGraw Hills, New Delhi.
5. HTML Black Book: Galgotia Publications.

Course Code:	D	C	A	P	5	0	1	Course Title:	**MODERN PROGRAMING TOOLS & TECHNIQUES -I**

WEIGHTAGE		
CA	**ETE (Pr.)**	**ETE (Th.)**
20	**20**	**60**

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction to Java**: Keywords, constants, variables and Data Types, Operators and Expressions, Control constructs, Introducing classes, objects and methods: defining a class, adding variables and methods, creating objects, constructors, class inheritance.
2.	**Arrays and String:** Creating an array, one and two dimensional arrays, string array and methods, Basics types, Classes and Objects: using super, Multilevel hierarchy abstract and final classes
3.	**Arrays and String:** Object class, Packages and interfaces, Access protection, Extending Interfaces, packages. Nested Classes, Inner Class Example Enum Types.
4.	**Classes:** String and String Buffer classes, Wrapper classes, Basics of Standard Java Packages (lang, util)
5.	**Exception Handling:** Fundamentals exception types, uncaught exceptions, throw, throw, Final, built in exception.
6.	**Multithreaded Programming:** Fundamentals, Java thread model: priorities, synchronization, messaging, thread classes, Runnable interface, suspending, resuming and stopping threads.
7.	**Input/Output Programming**: Basics, Streams, Byte and Character Stream, predefined streams, Reading and writing from files. Using Random Access Files.
8.	**Applets and AWT controls:** Meaning of Applet. AWT controls and Layout managers
9.	**Applets and AWT controls:** handling Images and sound. Basics of Swing Components and Layouts.
10.	**Event Handling**: The Event Delegation Model, Event Classes

LABORATORY WORK:

Sr. No.	Topics
1.	**Implementation Of JAVA Concepts**

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. E. Balaguruswamy, "Programming with Java: A Primer", Mc Graw Hill, Fourth Edition
2. Patrick Naughton and Herbertz Schildt, "Java-2 The Complete Reference", TMH,1999
3. The Java Handbook by Patrick Naughton, Michael Morrison Publisher: Osborne/McGraw-Hill
4. Horstmann, "Computing Concepts with Java 2 Essentials", John Wiley

Course Code:	D	C	A	P	4	0	7	Course Title:	**DATA STRUCTURE**

COURSE CONTENTS:

	WEIGHTAGE		
	CA	ETE (Pr.)	ETE (Th.)
	20	20	60

Sr. No.	Topics
1.	**Basic concepts and notations,** data structures and data structure operations
2.	**Complexity Analysis:** Mathematical notation and functions, algorithmic complexity and time space trade off, **Big O Notation**, The best, average & Worst cases analysis of various algorithms.
3.	**Arrays:** Linear & Multidimensional Arrays, Representation & traversal
4.	**Pointers**, Array Pointers, **Records** and Record Structures, Representation of Records in Memory; Parallel Arrays
5.	**Linked list:** representation, traversal, searching, Insertion, deletion of linked list. **Two way / multi linked structures,** Header Lists, Circular Lists
6.	**Stacks:** Basic operation of Stack, Memory Representation, Traversal. **Queues:** Operations, Representation & Types.
7.	**Recursion:** Definition, Function Call & Recursion implementation, Anatomy of Recursive Call, Complexity issues
8.	**Trees:-** Definition, Representation in memory.
9.	**Binary trees:** Binary tree traversal, Insertion, Deletion & Searching
10.	**Binary Search Trees:** Search, Insertion, deletion **Intro to Heaps**

LABORATORY WORK:

    1. Implementation of Arrays, Linked Lists, Stacks, Recursion, Trees and Heaps using C/C++

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Seymour Lipschutz, "Schaum Outline Series", Tata McGraw Hill, New Delhi, Year of Publication: 2006.
2. Mark Allen Weises, Data Structures & Algorithmic Analysis in C, Pearson Education.
3. Adam Drozdek, Data Structure & Algorithms in C++. Thomson.
4. Kruse, Data Structures & Program design, Prentice Hall of India, New Delhi.
5. Tenenbaum, Augenstein, & Langsam, Data Structures using C and C++, Prentice Hall of India, New Delhi.
6. Sorenson and Tremblay : An Introduction to Data Structures with Algorithms.

Course Code:	D	C	A	P	5	0	2	Course Title :	**COMPUTER ORGANIZATION AND ARCHITECTURE**

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
1.	**Review of Basics of Digital Electronics:** Codes, logic gates, flip flops, registers, counters, multiplexer, demultiplexer, decoder, and encoder.
2.	**Integers Representation:** Signed Magnitude, 1s & 2s Complement) & Real numbers (Fixed point & Floating Point representation), Register Transfer and Micro operations: Register transfer language Bus & memory transfer, logic micro operation, shift micro operation, Arithmetic Logic Shift Unit
3.	**Basic Computer Organization:** Instruction codes, computer instructions, timing & control, instruction cycles
4.	**Memory reference instruction**, Input/output & interrupts, Design of basic computer **Control Unit:** Hardwired vs. micro programmed control unit, Control Memory, Address Sequencing, Micro program Sequencer
5.	**Central Processing Unit**: General register organization, stack organization, instruction format, Addressing Modes Data transfer & manipulation, program control, RISC, CISC.
6.	**Introduction to Parallel Processing:** Pipelining ,Instruction pipeline, RISC Pipeline, Vector Processing
7.	**Computer Arithmetic:** Addition, Subtraction, Multiplication & Division Algorithm(s), Decimal arithmetic units & Operations.
8.	**Input-Output Organization:** Peripheral devices, I/O interface, data transfer schemes, program control, interrupt' DMA transfer, I/O Processor
9.	**Memory Organization Concepts:** Cache & Virtual memory
10.	**Multiprocessors:** Characteristics, Interconnection Structures, Interprocessor Communication and synchronization

READINGS: SELF LEARNING MATERIAL.
ADDITIONAL READINGS:

1. Morris Mano, "Computer System Architecture", Prentice Hall, Year of Publication: 2007.
2. David A Patterson, Computer Architecture A Quantitative Approach, Pearson Education Asia.
3. P. Pal Choudhuri, Computer Organisation and Design, PHI, New Delhi, 1994.
4. J.P.Hayes, Computer System Architecture, Pearson Education Asia.
5. Ali leigh, System Architecture, South Wester Publishing Co., New Delhi.
6. Parallel Computers by Rajaram & Murthy, EEE .

Course Code:	D	C	A	P	5	1	1	Course Title:	**E-COMMERCE & E-BUSINESS**

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction: E-Commerce, E-Business.** Meaning & Concept, E-Commerce vs Traditional Commerce, Media Convergence Business applications & Need for E-Commerce, Business. Basics of E-Commerce: Network and electronic transactions today
2.	The internet environment for E- Commerce, B2B, B2C transactions, providers and vendors
3.	Electronic Data Interchange to E-Commerce: EDI, UN/EDIFACT Standard
4.	The internet & Extranet for E-Commerce, Identification & Tracking tools for E-Commerce, Overview of Internet Bandwidth & Technology Issues
5.	Security Concerns, Security Solutions – Symmetric & Asymmetric Cryptosystems, Digital Signatures, PKCS, Protocol for Secure messaging, key management, X.509 Certificates, SET protocols, E-Cash over the Internet.
6.	Business Process Reengineering: BPR Approach, Strategic Alignment Model, BPR methodology, Rapid Re Methadology & PRLC
7.	Legal issues – Paper Document vs Electronic Document, technology for authenticating electronic document, Laws for E-Commerce, EDI interchange agreement, Legal issues for internet Commerce, Cyber Security, Cyber Crimes
8.	Management of Change, E-commerce in India
9.	**Case Study:** Designing and building E-Commerce web site. Managing Products, Database, Shopping cart applications, Integrating mobile E-Commerce, Payment Gateways, Tracking Orders. Eg**:** Amazon.com, eBay.com.
10.	Computer Emergency response team –CERT in objectives, functions, role , CERT –In Activities

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. K. Bajaj, "E-Commerce", Tata Mcgraw Hill, 1999
2. Sartaj Singh & Sandeep Kaur, E-Commerce E-Business: ABS
3. Jaynice Reynold & Roya Mofajali, "The Complete E-Commerce Book : Design, Build & Maintain a Successful Web-based Business"
4. Don Jones, Mark Scott & rick Villars, "E-Commerce for Dummies" Published by Hungry Minds
5. Ravi Kala Kota and Marcia Robinson : E-Business-Roadmap for Success; Pearson Education
6. Keneth C Laudon, Carol G Travor, "E-Commerce: business. technology. Society"

Course Code:	D	C	A	P	5	0	4	Course Title :	COMPUTER GRAPHICS

COURSE CONTENTS:

WEIGHTAGE		
CA	ETE (Pr.)	ETE (Th.)
20	20	60

Sr. No.	Topics
1.	**Fundamentals Of Computer Graphics:** Introduction, Overview of Image Representation, RGB Color Model, Display Monitor, Printer ,Mandelbrot Set ,Julia Set
2.	**Overview of Graphics I/O Devices:** Random scan displays, Raster scan displays, Interactive devices, Logical functioning of I/O devices, Output devices, Frame Buffer, Persistence, Resolution
3.	**Scan-Conversion:** Scan converting a Line, circle, ellipse, arcs & sectors, polygon, Region filling, Aliasing Effects, Antialiasing
4.	**2D Transformations**: Scaling, Translation, Rotation, shearing & reflection, Homogeneous Coordinate System, Composite Transformations
5.	**Two Dimensional Viewing:** Concept of Window, Viewport, Window to viewport Mapping, Graphic Pipeline, Panning, Zooming.
6.	**Clipping:** Point Clipping, Line Clipping - Cohen Sutherland, Midpoint subdivision, Liang-Barsky algorithm
7.	**Polygon Clipping:** Sutherland Hodgeman, Weiler Atherton **Projection:** Perspective Projection ,Parallel Projection
8.	**Hidden Surfaces: :** Z-Buffer, Scan Line, Back face, Painter's Algorithm, Area Subdivision
9.	**Color and Shading Model:** Light and Color, Phong Model, Interpolative shading Methods
10.	Texture, Ray Tracing, Additional Visual Effects, Animation, Morphing

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Zhigang Xiang, Roy Plastock, "Computer Graphics (Special Indian Edition) (Schaum's Outline Series)", Tata McGraw Hill, Second Edition
2. Author: Hearn and Baker, Title: Computer Graphics, Publishers: Pearson Education, Delhi Year of Publication: 2007
3. J.D. Foley, A.V. Dam, S.K. Feiner, J.F. Hughes, R.L. Phillips. "Introduction to Computer Graphics", Addison-Wesley Publishing company, N.Y.; Second Edition, 1994.
4. R.A. Plastock and G. Kalley, "Computer Graphics", Schaum Series McGraw Hill, 1986.

Course Code:	D	C	A	P	5	0	5	Course Title:	**MODERN PROGRAMMING TOOLS & TECHNIQUES – II**

WEIGHTAGE		
CA	**ETE (Pr.)**	**ETE (Th.)**
20	**20**	**60**

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction:** What is C#, Why C#, Evolution of C#, Characteristics of C#, Difference of C# from C++ and Java, Writing a C# program
2.	**Variables and Data Types:** Variables and Data Types, Boxing & Un-boxing, Operators & Expressions Decision Making and Looping: If, If else if. While, do while, for loop.
3.	**Handling Arrays:** Declaring Arrays. System.Array class **Methods:** Declaring Methods, Invoking Methods, Pass by Value, Pass by Reference
4.	**Classes & Object in C#:** Defining Classes, object, methods. Constructors, Using Polymorphism, Inheritance in classes.
5.	**Interfaces:** Meaning and Implementation **Namespaces:** Meaning and its working. Using System Namespace and Object class.
6.	**Exception Handling:** Exceptions, Multiple Catch Statements, Using Finally Statement, Nested Try Blocks
7.	**Windows Programming:** Using Controls- textboxes, listbox, buttons, datetime picker, comboboxes etc **Common Dialog Boxes:** OpenFileDialog, SaveFileDialog, ColorDialog, MessageBox Class and DialogResult Class.
8.	**File Input Output:** Working with Files and Directories. System.IO.
9.	**ADO.NET:** Accesing Database with ADO.NET. Executing Insertion, deletion, updation and select command with databases.
10.	XML Basics: What is XML? Data Representation through XML. Working with XMLReader and XMLWriter Classes.

LABORATORY WORK:

1. Implementation of all the concepts covered in syllabus.
2. Classes, methods.
3. Constructors, Polymorphism, Inheritance.
4. Namespaces, Collections.
5. Windows Programming.
6. File I/O.
7. ADO.NET, XML Basics.

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Programming in C# by E. Balagurusamy, Third Edition
2. Samuel J. Leffler Marshall Kirk McKusick Michael J. Karels John S. Quarterman, The programming with C#, Addison Wesley
3. C# Unleashed, Pearson Education
4. C# Black Book, Wiley
5. Beginning C# 2005 by Watson,Skinner, Publisher :Wiley. Year of Publication: 2005

Course Code:	D	C	A	P	5	0	6	Course Title :	**ARTIFICIAL INTELLIGENCE**

WEIGHTAGE	
CA	ETE (Th.)
20	80

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction and Overview**: Meaning Of AI, The AI Problems, Task Domains, AI Technique, Criteria for Success.
2.	**Problems, Problem Spaces & Search**: Defining The Problem As a State Space Search, Production Systems – BFS, DFS, Heuristic Search, Problem & Production System Characteristics, Issues In The Design Of Search Programs ,Common AI Problems
3.	**Heuristic Search Techniques:** Generate & Test, Hill Climbing, Best First Search, Constraint Satisfaction, Means-End Analysis
4.	**Knowledge Representation**: General Concepts Of Knowledge, Approaches of Knowledge Representation, Predicate Logic To Represent Knowledge, Resolution, Unification algorithm
5.	**Knowledge Representation using Rules**: Procedural Vs Declarative Knowledge, Logic Programming, Forward Vs Backward Reasoning, Matching & Control Knowledge
6.	**Symbolic Reasoning Under Uncertainty** - Nonmonotonic Reasoning **Statistical Reasoning** - Probability & Bayes Theorem, Certainty Factors and Rule Based Systems, Bayesian N/W, Fuzzy Logic and applications
7.	**Weak Slot And Filler Structures** :Semantic Nets, Frames **Strong Slot And Filler Structures:** Conceptual Dependency, Scripts
8.	**Natural Language Processing** – Introduction, Steps, Syntactic Processing, Semantic Analysis, Discourse & Pragmatic Processing, Spell Checking
9.	**Learning**: Meaning, Rote Learning, Learning by taking Advice, Learning from examples, Explanation-Based learning, Expert Systems & Its Architecture, Speech Recognition
10.	**Prolog:** Introduction, Converting English to Prolog Facts and Rules, Goals, Prolog Terminology, Variables, Control Structures, Arithmetic operators, Matching, Backtracking, Lists, Input/Output and Streams

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Rich, Knight, Nair "Artificial Intelligence", Tata McGraw Hill, Third Edition
2. D.W.Patterson, Introduction to AI & Expert Systems, Prentice Hall, 1990
3. P. H.Winston:, "Artificial Intelligence"
4. N.J.Nilsson, Principles of Artificial Intelligence, Kaufmann, 1980
5. Charnmiak & M. Dermalt, Introduction to AI , Addison Wesley, 1985.
6. A.J. Gongalez & D.D. Dankel, The Engineering of Knowledge based systems theory & practice, Prentice Hall, 1993.
7. G.F.Lager & W.A. Stubblefield, Artificial Intelligence and the design of Expert System , Benjamin Kummings, 1989.

Course Code:	**D**	**C**	**A**	**P**	**5**	**0**	**7**	Course Title:	**SYSTEM SOFTWARE**

	WEIGHTAGE	
CA	**ETE (Th.)**	
20	**80**	

COURSE CONTENTS:

Sr. No.	Topics
1.	**Introduction to System Software:** Definition, System software, Machine structure, Components of a programming system, Assemblers, linker, loader, compiler, Macros. **Evolution of Operating system:** Operating system architecture, User function of operating system, Batch control language, OS User Viewpoint: Facilities **Note:** Students should be allocated programs which can simulate the working of system programs and can be implemented using C / C++ programming.
2.	**Machine Structure and Assemblers Basic Functions**:- Machine structure, Approach to new machine, machine language, Assembly language
3.	**Design of assembler**: Design of assembler, Data structure, Format of databases, Algorithm, look for modularity. Table processing: Linear Search, Binary Search, Sorting, Hash Searching
4.	**Macro Language** :- Macro instructions, Features, Implementation
5.	**Introduction to Linking & Loading:** Loader Schemes – Compile and Go Loaders, General loader scheme, Absolute Loaders, Subroutine linkage, relocating loaders, direct linking loaders, Other loader schemes – Binders, Linkers, Loaders, Overlays, Dynamic Binders
6.	**Design of Absolute loader**, Design of Direct-linking loader – problem specification, data structures, format of databases **Note:** Students should be allocated programs which can simulate the working of system programs and can be implemented using C / C++ programming.
7.	**Programming Languages Concept:** Importance of high level languages, features, data types and data structure, storage allocation, Accessing of pointers and label variables
8.	**Programming Languages Concept: F**unctional, modularity, Asynchronous operation – conditions, signals, multitasking.
9.	**Formal Systems and Programming Languages:** Uses of formal systems in programming languages, Formal Specification, Formal Grammars
10.	**Formal Systems:** Hierarchy of Languages, Backus-Naur Form – Backus Normal Form – BNF, Canonic systems – Syntax specification, specification of translation, recognition and translation algorithm, Canonic systems and formal systems

READINGS: SELF LEARNING MATERIAL.

ADDITIONAL READINGS:

1. Author: John J. Donovan, Title: Systems programming, Publishers: Tata McGraw Hill
2. Dhamdhere, "Systems Programming and Operating Systems", TMH
3. Aho A.V. and J.D. Ullman, "Principles of Compiler Design", Addison Wesley
4. O.G. Kakde , "Compiler Design", Laxmi Publications.
5. Milan Milenkovic , "Operating Systems " Tata McGraw-Hill.

Course Code:	D	C	A	P	5	1	2	Course Title:	**WAP & WML**

WEIGHTAGE		
CA	ETE (Pr.)	ETE (Th.)
20	20	60

COURSE CONTENTS:

Sr. No.	Topics
1.	Understanding WAP, Introducing WML: what is WML, Getting started with WML? Understanding WML usage
2.	Writing for WAP in WML: Creating first card, Building decks of cards, Using basic navigation
3.	Card Navigation: Using URLs, Tags used in Navigation, Using phone buttons and Function keys, Navigation History.
4.	Managing Output: Basic Card output, Layout, Rendering
5.	Using Images: using images and icons, Image restrictions, Using localscr Images, Using images efficiently
6.	Working with user input: Using variables, Free form input with <input>, restricted input with <select>, under implemented input behavior, delivering data to applications
7.	Email Integration: Email and WAP, integrating Email into your applications
8.	Using WMLScript: WMLScript vs Javascript, Using WMLScript Functions, Using WMLScript Libraries, Sample Application
9.	Securing Applications: Security basics, WAP security architechture, Session management, WML for secure applications
10.	Writing for HTML and WML : why two languages, How to write for both languages, database driven applications, other languages

LABORATORY WORK:

Sr. No.	Topics
1.	Introduction to WML
2.	Building deck of cards using basic navigation.
3.	Use various tags in navigation along with phone keys and function keys
4.	Implementing images and icons, Image restrictions, Using localscr Images etc
5.	Defining user variables and perform operations on variable as
6.	Integrating email into any application to have developed
7.	Develop an application using WML script
8.	Perform various security applications

READINGS: SELF LEARNING MATERIAL.
ADDITIONAL READINGS:

1. Ben Farta, "WAP Development with WML and WMLScript", SAMS TechMedia, Year of Publication: 2006
2. Author: Title: 'Beginning WAP: Wireless Markup Language & Wireless Markup Language, Script ' Publisher: Wrox, Year: 2000.

| Course Code: | **D** | **C** | **A** | **P** | **6** | **0** | **6** | Course Title: | **BUSINESS INTELLIGENCE** |

WEIGHTAGE	
CA	ETE(Th.)
20	80

COURSE CONTENTS:

S. No.	Topics
1.	**Business Intelligence:** Introduction, Meaning, Purpose and Structure of Business Intelligence Systems. Understanding Multidimensional Analysis Concepts: Attributes, Hierarchies and Dimensions in data Analysis.Understanding Dimensional Data warehouse: Fact Table, Dimension Tables, Surrogate Keys and alternative Table Structure. What is multi dimension OLAP?
2.	**Understanding OLAP:** Fast response, Meta-data based queries, Spread sheet formulas. Understanding Analysis Services speed and meta data. Microsoft's Business intelligence Platform. Analysis Services Tools. Data Extraction, Transformation and Load. Meaning and Tools for the same.
3.	**Creating your first Business Intelligence Project.** Creating Data source, Creating Data view. Modifying the Data view. Creating Dimensions, Time, and Modifying dimensions. Parent-Child Dimension.
4.	**Creating Cube:** Wizard to Create Cube. Preview of Cube. Adding measure and measure groups to a cube. Calculated members. Deploying and Browsing a Cube.
5.	Advanced Measures and Calculations: Aggregate Functions. Using MDX to retrieve values from cube. Calculation Scripting. Creation of KPI's.
6.	**Advanced Dimension Design:** Creating reference, fact and many to many dimensions. Using Financial Analysis Cubes. Interacting with a cube. Creating Standard and Drill Down Actions.
7.	**Retrieving data from Analysis Services**: Creating Perspectives, MDX Queries, Excel with Analysis Services.
8.	**Data mining:** Meaning and purpose. Creating data for data mining. Data mining model creation. Selecting data mining algorithm. Understanding data mining tools. Mapping Mining Structure to Source Data columns. Using Cube Sources. Configuring Algorithm parameters.
9.	Creating Data mining queries and reports. Creation of Prediction queries. Understanding DMX language.
10.	Reporting Tools: Using SQL Server Reporting Services to develop reports for analysis services.

READINGS: SELF LEARNING MATERIAL
ADDITIONAL READINGS:
1. "Microsoft SQL Server 2008 Analysis Services",Scott Cameron.Microsoft Press. (2009)
2. "SQL Server 2008 Business Intelligence Development and Maintenance", Erric Veerman. Microsoft Press (For Data Mining only)
3. Business intelligence a managerial approach. Turban E, Sharda R, Aronson J.E. and King D.(2007). Prentice Hall
4. Mike Biere, Business Intelligence for the Enterprises, Prentice Hall, 2003.
5. Larissa T. Moss and Shaku Atre, Business Inteligence Roadmap: The complete Project Lifecycle for decision support Application, Addison-Wessly 2003.
6. Decision support and Data Warehousing systems Mallach E.G(2000). McGraw Hill.