## COURSE STRUCTURE WITH CREDIT

<table>
<thead>
<tr>
<th>Course</th>
<th>Marks</th>
<th>Hour/week</th>
<th>Credit</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA-101</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-102</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-103</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-104</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td>28</td>
</tr>
<tr>
<td>CA-105</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-I</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-II</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Marks</th>
<th>Hour/week</th>
<th>Credit</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA-201</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-202</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-203</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-204</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td>29</td>
</tr>
<tr>
<td>CA-205</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-III</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-IV</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>Seminar</td>
<td>25</td>
<td>01</td>
<td>01</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Marks</th>
<th>Hour/week</th>
<th>Credit</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA-301</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-302</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-303</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-304</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td>28</td>
</tr>
<tr>
<td>CA-305</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-V</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-VI</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course</th>
<th>Marks</th>
<th>Hour/week</th>
<th>Credit</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA-401</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-402</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-403</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-404</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td>28</td>
</tr>
<tr>
<td>Course</td>
<td>Credits</td>
<td>Theory</td>
<td>Practical</td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
<td>--------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>CA-405</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-VII</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-VIII</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-501</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-502</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-503</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-504</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA-505</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-IX</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>CA LAB-X</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>Softskill</td>
<td>75</td>
<td>02</td>
<td>03</td>
<td></td>
</tr>
</tbody>
</table>

**Semester-V**  
**Development Course**  

**Semester-VI**  
**Industrial Training/Project**  

31  
12
NORTH MAHARASHTRA UNIVERSITY, JALGAON
Department of Computer Science

Proposed SYLLABUS for Master of Computer Application (MCA)
(With Effect from June-2010)

Semester- I

CA-101 Computer Organization & Architecture
CA-102 Linux Operating System
CA-103 Mathematical Foundations of Computer Science
CA-104 C++ Programming
CA-105 System Programming
CA-Lab-I Lab on C++ Programming
CA-Lab-II Lab on Linux Operating System

Semester- II

CA-201 Object Oriented Analysis and Design (OOAD)
CA-202 Data Structures and Algorithms
CA-203 Operating Systems
CA-204 Database Management System (DBMS)
CA-205 Computer Networks
CA-206 Seminar
CA-Lab-III Lab on Data Structures
CA-Lab-IV Lab on DBMS

Semester- III

CA-301 Accounting and Management Control
CA-302 Design and Analysis of Algorithms
CA-303 Automata Theory and Computability
CA-304 Artificial Intelligence
CA-305 Java Programming
CA-Lab-V Lab on Design and Analysis of Algorithm
CA-Lab-VI Lab on JAVA Programming
Semester-IV

CA-401 Distributed Computing
CA-402 Internet Computing
CA-403 Network Programming
CA-404 Computer Graphics
CA-405 Optimization Algorithms
CA-Lab-VII Lab on Computer Graphics & Internet Computing
CA-Lab-VIII Lab on Linux Socket Programming

Semester –V

CA-501 Compiler Construction
CA-502 Software Engineering
CA-503 Current Computing Trends-I
CA-504 Windows and Visual C++ Programming
CA-505 Current Computing Trends-II
CA-506 Soft Skills Development
CA-Lab-IX Lab on Windows and Visual C++ Programming
CA-Lab-X Lab on Current Computing Trends

Semester-VI

CA-601 Full time Industrial Training
CA-101: Computer Organization & Architecture

1. Digital Logic Circuits and Components: [6]
   Sequential Circuits: Flip flops, Registers, Shift Registers, Binary Counters - Memory Unit.

2. Data Representation : [4]
   Data Types - Complements - Fixed Point Representation - Floating Point Representation - Other
   Binary Codes - Error detection Codes

3. Processor Organisation : [8]
   General Register Organisation - ALU - Instruction codes - Instruction Formats - Stack
   Organisation - Addressing modes

4. Control Unit : [8]
   Register transfer and micro operations, Timing and Control, Control Memory, micro
   programming, Hard wired control

5. 8085 Microprocessor : [8]
   Internal Architecture, Instruction Set, Assembly Language programming

6. Input/Output Organisation : [8]
   I/O interface, Asynchronous data transfer, Modes of transfer, priority Interrupt, Direct memory
   access.

7. Memory Organisation: [8]
   Memory Hierarchy, Main memory, Auxiliary memory, Associate Memory, Cache Memory, and
   Virtual memory.

BOOKS:

1) Computer System Architecture , M. Morris Mano, Prentice Hall of India Pvt. Ltd., Eastern
   Economy Edition.

2) Micro processor Architecture, Programming & Applications with the 8085, Ramesh S
   Goankar, Penram International Publishing (India) Pvt. Ltd.


CA-102: Linux Operating System

1. History and Development of Linux [4]

2. User Accounts [2]

3. Introduction to the Filesystem & Linux Permissions [6]

   Using the vi Editor, Using Other Editors, Examining File Contents, Redirection. Archiving Files with tar, Archiving Files with cpio, Archiving Files with Other Utilities, Zipping Files, Examining Backup Issues.


   Configuring a Local Printer, Printing, Managing Print Spools and Queues, Configuring Remote Printers.

7. Package Management & Configuring the Linux Environment [10]
   Examining Package Solutions, Managing Packages with RPM, Verifying and Repairing Applications, Upgrading and Freshening Packages. Examining Shells, Using Variables, Examining Linux Configuration Script Files, Examining System Startup Files, Examining the /etc/fstab File, Examining the cron System, Creating a Shell Scripts.

Managing Jobs and Background Processes, Using the Process Table to Manage Processes, Introducing Delayed and Detached Jobs.

9. Configuring and Managing Services
Starting and Stopping Services, Identifying Core and Non-critical Services, Configuring Basic Client Services, Configuring Basic Internet Services, Working with Modules.

10. Managing Network Services

BOOKS:

7. Peterson, Linux: Complete Reference (5th Ed), Peterson, TMH.

CA-103 Mathematical Foundations of Computer Science

1. Logic:
Prepositional Equivalences, Truth tables, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.

2. Relations:
Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations, Closures of Relations, Equivalence Relations, Partial Orderings.

3. Functions:
One-to One and Onto Functions, Inverse Functions and Compositions of Functions, Graphs of Functions, Some Important Functions.
4. Graphs: [12]

Introduction to Graphs and Graph Models, Terminology and Special Types of Graphs, Representations of Graphs, Isomorphism, Connectivity, Euler and Hamiltonian Paths - Shortest Path problems - Planar Graphs - Graph Coloring.

5. Trees: [12]

Introduction to Trees, Applications of Trees, Traversals, Spanning Trees, Minimum Spanning Trees.

6. Algebraic Structures: [8]


BOOKS:


CA-104: C++ Programming

1. Introduction: [4]
What is object-oriented programming? Why Do We Need Object-Oriented Programming characteristics of Object-Oriented Languages. C++ and C.

Output Using cout. Directives. Input With cin. Type bool. The setw Manipulator. Type Conversions.

3. Functions: [4]
4. Object and Classes: [6]

5. Arrays and String: [4]


Concept of Inheritance, Derived Class And Base Class, Derived Class Constructors, Overriding Member Function, Inheritance In The English Distance Class, Class Hierarchies, Inheritance And Graphics Shapes, Public And Private Inheritance, Levels Of Inheritance, Multiple Inheritance, Ambiguity In Multiply Inheritance, Aggregation: Classes Within Classes, Inheritance And program Development.

Addresses And pointer, The Address-Of Operator &, Pointer And Arrays, Pointer And Faction, Pointer And C- Types String, Memory Management: New And Delete, Pointers To Objects, Debugging pointers.

Virtual Function, Friend Function, Static Function, Assignment and Copy Initialization, This Pointer, Dynamic Type Information.

10. Streams and Files: [4]
Streams Classes. Stream Errors. Disk File I/O with Streams, File Pointers, ErrorHandling In File I/O File I/O With Member Function, Overloading the Extraction And Insertion Operators Memory As A Stream Object, Command line Arguments, and Printer Output.

11. Templates And Exceptions: [4]
Function Templates, Class Templates Exceptions.


BOOKS:


3. The Complete Reference c- - By Herbert Sehlidt Tata Megraw-hill publication.


5. OOPS C++ Big C++ Cay Horstmann Wiley Publication

**CA-105: System Programming**

1. **Background:**
   Introduction to grammars, languages, finite state machines.

2. **Introduction to Systems Programming:**
   Introduction to Assembly Language Programming - Introduction to Instruction Formats, Data formats - Role of Base Register, Index Register.

3. **Assembler:**
   Introduction to Assembler, databases used in assembler design, Design of Assembler - Single Pass & Double Pass.

4. **Macro Processor:**

5. **Loaders:**
   Introduction to Loaders, functions of a loader, types of Loaders, databases used in Loaders, Design of Loaders - Absolute & DLL.

6. **Compilers:**
   Introduction to compilers: a brief discussion on various phases of compilers. Applications of FSM and grammars in compiler design

7. **Software Tools:**
   Introduction to Software Tools, Text editors, Interpreters, Program Generators, Debug Monitors.

**BOOKS:**


CA-Lab-I Lab on C++ Programming

1. Write a program to demonstrate encapsulation using of class
2. Write a program to demonstrate use of constructor, constructor overloading and destructor
3. Write a program to demonstrate use of array manipulations
4. Write a program to demonstrate use of string manipulations
5. Write a program to demonstrate use of function overloading
6. Write a program to demonstrate use of operator overloading
7. Write a program to demonstrate use of friend class
8. Write a program to demonstrate use of friend function
9. Write a program to demonstrate use of recursive function
10. Write a program to demonstrate use of array of objects
11. Write a program to demonstrate use of pointers
12. Write a program to demonstrate use of pointer arithmetic
13. Write a program to demonstrate use of operator overloading
14. Write a program to demonstrate use of all types of inheritance
15. Write a program to demonstrate use of function templates
16. Write a program to demonstrate use of class templates
17. Write a program to demonstrate use of formatted I/O operation
18. Write a program to demonstrate use of unformatted I/O operation

CA-Lab-II Lab on Linux Operating System

Instructor should ask students to give live demonstrations on:

**System Access**: Logging In, Linux Commands, Getting Help, Obtaining Information About Your System, Logging In and Using Remote Linux Systems.

**Starting and Stopping Linux**: Shutting Down a Linux System, Booting a Linux System, Other Boot Methods.

**User Accounts**: Creating Additional User Accounts, Groups, Managing Users and Groups.
**Filesystem**: Filesystem Navigation, Managing the Filesystem, Performing Filesystem Maintenance, Locating Files.


**Creating and Viewing Files**: Using the *vi* Editor, Using Other Editors, Examining File Contents Redirection.

**Archiving Files**: Archiving Files with *tar*, Archiving Files with *cpio*, Archiving Files with Other Utilities, Zipping Files, Examining Backup Issues.

**Shell Scripts**: Creating a Shell Scripts (Create at least ten shell scripts).


**Printing Files**: Configuring a Local Printer, Printing, Managing Print Spools and Queues Configuring Remote Printers.

**Configuring the Linux Environment**: Examining Shells, Using Variables, Examining Linux Configuration Script Files, Examining System Startup Files, Examining the `/etc/fstab` File, Examining the cron System.

**Multitasking**: Managing Jobs and Background Processes, Using the Process Table to Manage Processes, Delayed and Detached Jobs.

**Configuring and Managing Services**: Starting and Stopping Services, Identifying Core and Non-critical Services, Configuring Basic Client Services, Configuring Basic Internet Services Working with Modules.


**Preparing for a Linux Installation**: Gathering Installation Information, Creating Installation Disks, Partitioning the Disk.

**Installing Red Hat Linux** (or any other distribution)
Installing Red Hat Linux, Configuring Linux.
BOOKS:
7. Peterson, Linux: Complete Reference (5th Ed), Peterson, TMH.
CA-201: Object Oriented Analysis and Design (OOAD)

1. Introduction: [8]

2. Methodology and UML: [10]

3. Object Oriented Analysis: [12]
   Identifying Use case – Business object analysis – Use case driven object oriented analysis – Use case model – Documentation – Classification – Identifying object, relationships, attributes, methods – Super-sub class – A part of relationships Identifying attributes and methods – Object responsibility

4. Object Oriented Design: [12]
   Design process – Axions – Colollaries – Designing classes – Class visibility – Refining attributes – Methods and protocols – Object storage and object interoperability – Databases – Object relational systems – Designing interface objects – Macro and Micro level processes – The purpose of a view layer interface

5. Software Quality: [8]

BOOKS:


CA-202: Data Structures and Algorithms

1. Introduction to Data Structures: [6]
   Abstract Data Types, Review of strings, multi-dimensional arrays, structures and pointers concepts in C/C++.

2. Stack: [6]
   Specification of ADT and primitive operators, Representing Stacks in C/C++, Applications of Stacks: Infix, Postfix and prefix expression handling.

   Recursion Definition and Processes, Recursion in C, Writing Recursive Programs, Simulating Recursion, Efficiency of Recursion.

4. Queues and Lists: [8]
   The queues and its Sequential Representation, Linked lists, Lists in C/C++, Circular Linked lists, Doubly linked lists.

5. Trees: [8]
   Binary Trees, Binary Tree Representations, Trees and their Applications, Searching: Basic Search Technologies, Tree Searching,

6. Graphs and Their Applications: [8]
   Graphs, Graph Traversal and Spanning Forests, Prim's algorithm.

7. Sorting: [8]
   General Background, Exchange Sorts, Selection and Tree Sorting, Insertion Sorts, Merge and Radix Sorts.

BOOKS:


2. Data Structures Using 'C' by Bala Guruswamy, TMH

3. Data Structures Using 'C' by Weiss, Pearson Education


CA-203 Operating Systems

1. Introduction to OS: [4]
   What is an Operating System?, What are the components of an OS?, What does an OS do? Short History of Oss, Different types of OS.
2. **Computer System Structures:** [3]


3. **Operating System Structures:** [4]

   System Components, Operating System Services, System Calls, System Programs, System Structure, Virtual Machines, System Design and Implementation.

4. **Processes:** [4]


5. **Threads:** [2]

   Overview, Multithreading Models, Thread Libraries, Thread Pools.

6. **CPU Scheduling:** [4]

   Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-Processor Scheduling, Real-Time Scheduling, Algorithm Evaluation.

7. **Process Synchronization:** [6]

   Background, The Critical-Section Problem, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors.

8. **Deadlocks:** [4]

   System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock, Combined Approach to Deadlock Handling.

9. **Memory Management:** [5]

   Background, Address Binding - Linking and Loading, Swapping, Contiguous Allocation Paging, Segmentation, Segmentation with Paging.

10. **Virtual Memory:** [4]

    Background, Demand Paging, Process Creation, Page Replacement, Allocation of Frames, Thrashing.

11. **File System:** [3]
FILE CONCEPT, ACCESS METHODS, DIRECTORY STRUCTURE, FILE-SYSTEM MOUNTING, FILE SHARING, PROTECTION.

12. FILE SYSTEM STRUCTURE:

File structure: Logical storage unit—Collection of related information, File system resides on secondary storage (disks), File system organized into layers, File control block – storage structure consisting of information about a file.

13. MASS STORAGE:

Overview of Mass Storage, Disk Mechanism, Disk Structure, Disk Scheduling.

BOOKS:


3. Operating System, Achut Godbole, TMH.


CA-204 DATABASE MANAGEMENT SYSTEM (DBMS)

1. Introductory Concepts

Databases and Information Systems, An example usage context, Database system concepts and architecture.

2. Semantic Database Design

High-level conceptual modeling, ER Modeling concepts, ER Diagrams, Cardinality constraints, Higher-order relationships, Enhanced ER Model (EER), Weak-entity types, Subclasses and inheritance, Specialization and Generalization, Modeling of UNION types using categories.

3. Relational Model, Languages and Systems
Relational algebra: Relational model concepts, Relational integrity constraints, Update operations on relations, Relational algebra model, ER to relational mapping

SQL: Data definition in SQL, Queries and update statements, Views, Integrity constraints, Specifying indexes, Embedded SQL.

IBM DB2 case study: Architecture of DB2, Data definition and manipulation in DB2.

EER to Relational mapping.

4. Database design using the relational model
[6]

Functional dependencies: Keys in a relational model, Concept of functional dependencies, Normal forms based on primary keys, Boyce-Codd Normal Forms

Further Dependencies: Multi-values dependencies and fourth normal form, Join dependencies and fifth normal form, Inclusion dependencies, Other dependencies and normal forms

5. Storage and Indexing Structures
[6]

Storage structures: Secondary storage devices, Buffering of blocks, File Organization, Heaps, Sorted Files, Hashing and overflow handling techniques, Dynamic hashing, Extensible hashing, Other file organizations

Indexing methods: Basic terminology, Primary indexes, Clustering index, Secondary index, Multilevel indexes, ISAM, B-trees, B+ trees, inserting and searching algorithms for B+ trees, Other indexing methods.

6. Transaction Processing and Concurrency Control
[6]

Transaction Fundamentals: OLTP environments, Concurrency issues, need for transactions, Necessary properties of transactions (ACID properties), Transaction states, serializability, Serial schedules, Conflict serializability, View serializability, Recoverable and non-recoverable schedules, Cascading rollbacks, Cascadeless schedules.

Concurrency control: Serialized and non-serialized schedules, Testing for serializability, Locking, Lock compatibility matrix, Locking and serializability, Deadlocks and starvation, Two-phase locking (2PL) protocol, Conservative, strict and rigorous 2PL, 2PL with lock conversions, Timestamp-ordering based protocol, Multi-versioning protocol, Multi-granularity locking, Deadlock prevention protocols, Wait-die and wound-wait schemes, Time-out based schemes, Deadlock recovery, Nested transactions.

7. Database recovery techniques
[4]
Recovery concepts, Deferred updates technique, Immediate update technique, Shadow paging, ARIES recovery algorithm.

8. **Query Processing and Optimization** [4]

Translating SQL into relational algebra, Basic query operations, Heuristics in query optimization, Selectivity and cost estimates in query optimization, Semantic query optimization


Discretionary access control, Mandatory access control and multi-level security, Statistical database security

10. **Enhanced Data Models for specific applications** [4]

Active database concepts, Temporal databases, Spatial databases, multi-media databases

11. **Distributed databases and issues** [4]

Data fragmentation, replication and allocation in distributed databases, Types of distributed database systems Query processing in distributed databases, Concurrency control and recovery in distributed databases

**BOOKS:**


**CA-205: Computer Networks**

1. **Introduction:** [4]
Data communications, Networks, The Internet, Protocol & Standards

2. **Network Models:** [4]
Layered tasks, Internet model, OSI model

3. **Physical Layer:** [10]
Analog Transmission: Modulation digital data, telephone modem, Modulation analog signals.
Multiplexing: FDM, WDM, TDM.
Transmission Media: Guided media, unguided media.
Circuit Switching & Telephone Network: Circuit switching, telephone network.

4. Data Link Layer: [10]
Error detection and Correction: Type of errors, detection and correction of errors.
Data Link Control & Protocol: Flow & error control, Stop-And-Wait ARQ, Go-Back-N ARQ, Select Repeat ARQ, HDLC.
Point-To-Point Access: Point-to-point protocol, PPP stack.
Local Area Network: Traditional Ethernet, fast and gigabit Ethernets.
Connecting LANs, Backbone Networks and Virtual LANs: Connecting devices, Backbone networks, Virtual LANs.

5. Network Layer: [10]
Internetworks, Addressing, Routing.
Unicast routing, Unicast routing protocols, Multi routing, Multicast routing protocols.

Process-To-Process delivery, user data gram, Transmission control protocol.

Client-Server Model: Client-Server model, Socket interface.
A brief introduction to DNS, SMTP, FTP.

BOOKS:

CA-206 Seminar

1. The seminar shall be conducted as per the following depending on the applicability of the subject - Every week there shall be a period for seminar wherein the students have to select the topic, carry out literature survey and prepare seminar report.

2. No grade will be given for the seminar. However, the completion of the credit for the seminar shall be compulsory.
CA-Lab-III Lab on Data Structures

Implementation of application programs based on the following:

- Arrays
- Records, Structure
- Pointers
- Multidimensional Arrays, Matrices,
- Stacks, Polish Notation
- Queues
- Deques
- Linked List, Circular Link List, Doubly Link List

Implementation of programs for sorting techniques

- Bubble sort
- Selection sort
- Insertion sort
- Radix sort
- Quick sort
- Merge sort
- Heap sort

Implementation of programs based on Trees

- Binary Search Tree
- Heap Tree
- Balanced Binary Tree
- m-way search tree

Implementation of programs based on Graphs

- Depth First Traversal
- Breadth First Traversal
- Obtaining Shortest Path
- Minimum spanning tree
CA-Lab-IV Lab on DBMS

DBMS lab

Course Description: This course explores database programming using both native and embedded ANSI-standard Structured Query Language (SQL). Topics include enterprise database management systems, database middleware, data definition language, data manipulation language, data control language, database queries reporting, query optimization, and database views. Student assignments include database creation, query design and programming, and database manipulation via embedded SQL calls from a programming language.

Representative List:

1. Creating database tables and using data types.
   - Create table
   - Modify table
   - Drop table
2. Practical Based on Data Manipulation.
   - Adding data with Insert
   - Modify data with Update
   - Deleting records with Delete
3. Practical Based on Implementing the Constraints.
   - NULL and NOT NULL
   - Primary Key Constraint
   - Foreign Key Constraint
   - Unique Constraint
   - Check Constraint
   - Default Constraint
4. Practical for Retrieving Data Using following clauses.
   - Simple select clause
   - Accessing specific data with Where
   - Ordered By
   - Distinct
   - Group By
5. Practical Based on Aggregate Functions.
   - AVG
   - COUNT
   - MAX
   - MIN
   - SUM
   - CUBE
6. Practical Based on implementing all String functions.
7. Practical Based on implementing Date and Time Functions.
8. Practical Based on implementing use of UNION, INTERSECTION, SET DIFFERENCE.
9. Implement Nested Queries & all types of JOIN operation.
10. Practical Based on performing different operations on a view.
11. Practical Based on implementing use of Procedures.
12. Practical Based on implementing use of Triggers
13. Practical Based on implementing Cursor.
14. Make a Database connectivity with front end tools like – VB, VC++,D2K.
15. Practical based on creating Data Reports.
16. Design entity relationship models for a business problem and develop a normalized database structure
CA-301 Accounting and Management Control

1. Accounting:  
Principles, concepts and conventions, double entry systems of accounting, introduction to basic books of accounts of sole proprietary concern, closing of books of accounts and preparation of trial balance. Final accounts, training, Profit and Loss accounts and balance sheet of sole proprietary concern (without adjustment).

2. Financial management:  
Meaning, scope and role, a brief study of functional areas of financial management. Introduction to various FM tools: ratio analysis, fund flow statement, cash flow statement.

3. Costing:  
Importance and basic principles, a brief introduction to methods of costing and elements of cost. Marginal costing, nature, scope and importance, Break-even analysis, its use and limitations, construction of break even chart. Standard costing, nature and scope, computation and analysis of variances with reference to material cost, labour cost and overhead cost, interpretation of the variances.

4. Introduction to Management Control System:  
Goals, strategies, key variables, performance measures, responsibility centers and transfer price, investment center, reporting systems, management by objectives, budgeting and control, strategic and long range planning.

BOOKS:


CA-302 Design and Analysis of Algorithms

1. Introduction [8]

2. Divide and conquer [8]
Introduction, Binary Search, Sorting, Median Finding

3. Dynamic Programming [10]
Combinatorial Search:
Dynamic programming 1: Knapsack
DP 2: Longest common subsequence
DP 3: Matrix chain multiplication or Optimal search trees
DP 4: A machine scheduling problem

Introduction, Set of Intervals, Fractional Knapsack, Huffman Coding

5. NP-Completeness [14]
Matching, Introduction to NP-Complete, Search/Decision, SAT, Independent_Set 3VC, Exact Cover, Multi Set, Subset Sum & Partition, Hamiltonian Circuit

BOOKS:

2. Introduction to the design and analysis of Algorithms, Anany Levitin : Pearson Education,
CA-303: Automata Theory and Computability

1. Finite Automata: [12]
   Sets, relations, functions, graphs, trees, mathematical induction, Finite Automata (FA),
   definition, description, transition systems, acceptability of a string, NFA, DFA,
   equivalence of DFA and NFA, Melay Moore model, minimization of automaton,
   Applications.

2. Formal Languages: [04]
   Formal languages, Chomsky classification of languages, languages, their relation and
   automaton.

3. Regular Expressions: [14]
   Regular expressions, FA and regular expressions, pumping lemma for regular sets,
   applications of pumping lemma, closure properties of regular sets, regular sets and
   regular grammars.

4. Context Free Languages: [10]
   CFLs and derivation trees, ambiguity in Context-Free Grammars (CFGs), simplification
   of CFGs, Normal Forms for CFGs(CNF and GNF), pumping lemma for CFLs, decision
   algorithms for CFLs.

   Pushdown Automaton (PDA), informal description, basic definitions, acceptance by a
   PDA, PDA and CFLs.

   Turing Machine, Model, computable languages and function, representation of TMs,
   Language Acceptability by TMs, Design of TM, Halting Problem of TMs.

BOOKS:


2. J.E.Hopcraft, R. Motwani and J.D.Ullman, Introduction to Automata Theory languages &
   Computation, Pearson Education Asia.


CA-304: Artificial Intelligence

1. Introduction: [4]
   What is Artificial Intelligence?, The AI Problems, The Underlying Assumption, What is an AI Technique, The Level of the Model, Criteria for Success, Some General References, One Final Word.

   Defining the Problem as a State Space Search, Production systems, Problem Characteristics, Production System Characteristics, Issues in the Design of Search Programs, Additional Problems.

   Generate-and-Test, Hill Climbing, Best-First Search, Problem Reduction, Constraint Satisfaction, Means-Ends Analysis.


   Representing Instance and Isa Relationships, Computable Functions and Predicates, Resolution, Natural Deduction.


7. Symbolic Reasoning under Uncertainty: [4]

8. Statistical Reasoning: [10]
   Probability and Baye’s Theorem, Certainty Factors and Rule-Based Systems, Bayesian Networks, Dempster-Shafer Theory, Fuzzy Logic.
Semantic Nets, Frames.

Conceptual Dependency, Scripts, CYC.

11. Knowledge Representation Summary: [2]
Syntactic-Semantic Spectrum of Representation, Logic and Slot-and-Filler Structures, Other Representational Techniques, Summary of the Role of Knowledge.

BOOKS:

1. Artificial Intelligence, Elaine Rich, Kevin Knight, Tata McGrawHill.


CA-305 Java Programming

1. An Introduction to Java: [4]
Java As a Programming Platform, The Java "White Paper" Buzzwords, Java Applets and the Internet, A Simple Java Program, Comments, Data Types, Variables, Operators, Strings, Input and Output Control Flow

2. Objects and Classes: [6]
Introduction to Object-Oriented Programming, Using Predefined Classes, Defining Your Own Classes, Static Fields and Methods, Method Parameters, Object Construction, Packages

3. Inheritance, Interfaces and Inner Classes: [6]
Classes, Superclasses, and Subclasses, Object: The Cosmic Superclass, Generic Array Lists, Object Wrappers and Autoboxing, Methods with a Variable Number of Parameters, Enumeration Classes, Reflection, Interfaces, Object Cloning, Interfaces and Callbacks, Inner Classes, Proxies.

Introducing Swing, Creating a Frame, Positioning a Frame, Displaying Information in a Component, Working with 2D Shapes, Color, Special Fonts for Text, Displaying Images.

5. Event Handling and User Interface Components with Swing: [6]
6. Deploying Applications and Applets:
JAR Files, Java Web Start, Applets, Storage of Application Preferences

7. Exceptions, Logging and Assertions:
Dealing with Errors, Catching Exceptions, Tips for Using Exceptions, Using Assertions Logging

8. Multithreading
What AreThreads?, Interrupting Threads, Thread States, Thread Properties, Synchronization, Blocking Queues, Thread-Safe Collections, Callables and Futures, Executors, Synchronizers, Threads and Swing

9. Streams, Files and JDBC
Streams, Text Input and Output, Reading and Writing Binary Data, ZIP Archives, Object Streams and Serialization, File Management, New I/O Regular Expressions
The Design of JDBC, JDBC Configuration, Executing SQL Statements, Query Execution Scrollable and Updatable Result Sets, Row Sets, Metadata, Transactions

BOOKS:


3. Steven Holzner, JAVA 2 Programming Black Book, Wiley India.

4. Ivor Horton, Beginning Java 2, JDK 5 Ed, Wiley India.

CA-Lab-V Lab on Design and Analysis of Algorithm

1. Write a program for creating max./min. heap using
   • INSERT
   • ADJUST/HEAPIFY
2. Write a program to implement union and find operation.
3. Write a program to find minimum and maximum form a given array.
4. Write a program for searching element form given array using binary search for n=1000,2000,3000 find exact time of execution.
5. Write a program for sorting given array in ascending/descending order with n=1000,2000,3000 find exact time of execution using
   • Heap sort
   • Merge sort
   • Quick sort
6. Write a program for matrix multiplication using Strassen’s matrix multiplication.
7. Write a program to find solution of Knapsack instant.

8. Write a program to find shortest path using single source shortest path.
9. Write a program to find shortest path using all pair path.
10. Write a program to implement breadth first and depth first traversal.
11. Write a program to implement topological sort.
12. Write a program to implement breadth first and depth first traversal.
13. Write a program to find all solutions for 8-queen problem using backtracking.

CA-Lab-VI Lab on JAVA Programming

1. Write a program that demonstrate program structure of java.
2. Write a program that demonstrate string operations.
3. Write a program that demonstrate package creation and use in program.
4. Write a program that demonstrate inner class.
5. Write a program that demonstrate inheritance.
6. Write a program that demonstrate 2D shapes on frames.
7. Write a program that demonstrate text and fonts.
8. Write a program that demonstrate event handling for various types of events.
9. Write a program to illustrate multicasting.
10. Write a program to illustrate use of various swing components.
11. Write a program that demonstrate use of dialog box.
12. Write a program to create own dialog box.
13. Write a program to create toolbar, menu & popup menu.
14. Write a program to implement file handlings.
15. Write a program that demonstrate Applet programming.
16. Write a program to implement generic programming.
17. Write a program that demonstrate JDBC on applet/application.
18. Write a program that demonstrate multithreading.

Semester- IV

CA-401: Distributed Computing


Introduction, Examples of distributed systems, Resource sharing and the Web, Challenges.

2. System models: [2]
Introduction, Architectural models, Fundamental models.

3. Networking and Internetworking:

Introduction, Types of network, Network principles, Internet protocols, Network case studies: Ethernet, wireless LAN and ATM.

4. Interprocess communication:

Introduction, The API for the Internet protocols, External data representation and marshalling, Client-server communication, Group communication, Case study: Interprocess communication in UNIX.

5. Distributed Objects and Remote Invocation:

Introduction, Communication between distributed objects, Remote procedure call, Events and notifications, Java RMI case study.

6. Distributed File Systems:

Introduction, File service architecture, Sun Network file system, The Andrew File System, Recent advances.

7. Name Services:

Introduction, Name services and the Domain Name System, Directory and discovery services, Coordination and Agreement:

Case study of the Global Name Service, Case study of the X.500 Directory Service.

8. Time and Global States:

Introduction, Clocks, events and process states, Synchronizing physical clocks, Logical time and logical clocks, Global states, Distributed debugging.

Introduction, Distributed mutual exclusion, Elections, Multicast communication, Consensus and related problems.

9. Transactions and Concurrency Control:

Introduction, Transactions, Nested transactions, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control.

10. Distributed Transactions:
Introduction, Flat and nested distributed transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery.


Introduction, system model and group communication, fault tolerant services, Highly available services, Transactions with replicated data.

BOOKS:


CA-402 Internet Computing

(Asp.Net 3.5)

Unit I


Unit II

Data Binding in Asp.Net 3.5, Data Management with ADO.Net, Querying with LINQ, Site Navigation, Personalisation, Membership and Role Management, Portal Framework with Web Parts. [22]

Unit III


BOOKS:


CA-403 Network Programming

Unit I:

[10]

Unit II:
Client server Programming and Application: The client server model and software design, the socket interface, concurrent processing in client-server software, program interface to protocol algorithms & issues in client Software design, example client software, algorithms & issues in server software design Iterative connectionless server, iterative connection oriented server, single process Concurrent server concurrent connection oriented server, multiprotocol server, multi-service server concurrency in client external data representation remote procedure call concept, RPCgen concept, network file system (NFS) concepts DNS, TELNET DHCP, Electronic mail (822, SMTP, MIME), discussion on the topic “the feature of TCP/IP”.

[25]

Unit III:
Network Interface Layer: Overview of network interface layer media access control standards, mapping the Physical address to the IP address. Internet Layer: Purpose of the internet layer, classes of IpV4 addresses, basics of routing, IP datagram ICMP, IGMP Transport Layer Types of data transfer connection-less data transfer, connection-oriented data transfer.

[15]

BOOKS:


CA-404 Computer Graphics

1. Introduction to Computer Graphics: [4]


2. Two-Dimensional Transformations: [4]


3. Three-Dimensional Transformations: [4]


5. Scan conversion – lines, circles and Ellipses; Filling polygons and clipping algorithms: [6]
Scan Converting Lines, Mid-point criteria, Problems of Aliasing, end-point ordering and clipping lines, Scan Converting Circles, Scan Converting Ellipses, Filling Polygons, edge data structure, Clipping Lines algorithms– Cyrus-Beck, Cohen-Sutherland and Liang-Barsky, Clipping Polygons, problem with multiple components.


Representing Solids, Regularized Boolean Set Operations, Primitive Instancing, Sweep Representations, Spatial-Partitioning Representations - Octree representation, B-Reps, Constructive Solid Geometry, Comparison of Representations.


8. Illumination and Shading: [4]

Illumination and Shading Models for Polygons, Reflectance properties of surfaces, Ambient, Specular and Diffuse reflections, Atmospheric attenuation, Phong’s model, Gouraud shading, some examples.


Why OpenGL, Features in OpenGL, OpenGL operations, Abstractions in OpenGL – GL, GLU & GLUT, 3D viewing pipeline, viewing matrix specifications, a few examples and demos of OpenGL programs.

11. Miscellaneous topics: [4]

Why Realism?, Aliasing and Antialiasing, texture bump mapping, Animation methods, methods of controlling animation, soft modeling of objects, image based rendering, Fundamental Difficulties.

12. Image Manipulation and Storage: [4]

BOOKS:


CA-405 Optimization Algorithms

1. Overview of operations Research: [2]
   OR models – OR Techniques

2. Linear Programming: [8]

3. Dual problems: [8]
   Relation between primal and dual problems – Dual simplex method

4. Transportation model: [8]


   Recursive nature of dynamic programming – Forward and Backward Recursion

7. Deterministic Inventory Models : [6]
   Static EOQ Models – Dynamic EOQ models.
8. **Game theory:** Two person Zero Sum Games – Mixed strategy games and their algorithms.

**BOOKS:**


**CA-Lab-VII Lab on Computer Graphics & Internet Computing**

**Computer Graphics:**
Implement following algorithms (Representative List):

1. Line drawing algorithm
2. Circle drawing algorithm
3. Ellipse drawing algorithm
4. Polygon filling algorithm
5. Windowing and clipping algorithm (Point, line and polygon clipping)
6. Composite 2-D transformation, (rotation, scaling & reflection )
7. 3-D geometric transformation (rotation, scaling & reflection )
8. 3-D Curve and surface representation i.e. B-spline curves and Surfaces, polynomial curves and surfaces, Bezier curves and Surfaces.
9. Determination of visible surfaces and lines,
10. Combining all modules to form a single graphics package

**Internet Computing:**

1. Create and ASP .NET application using Web server controls apply appropriate validation to it.
2. Create an ASP .NET application using Master Pages and Themes and Skins
3. Create an ASP .NET application to demonstrate binding of GridView, DataList, Repeater, DropDownList, RadioButtonList, CheckBoxList Control using ADO .NET
4. Create an ASP .NET application to demonstrate LINQ to XML, LINQ to Objects, LINQ to SQL
5. Create an ASP .NET application to demonstrate Navigation controls
6. Create an ASP .NET application to demonstrate User and Role management.
7. Create an ASP .NET application to demonstrate User controls
8. Create an ASP .NET application to demonstrate Web parts
9. Create an ASP .NET application to demonstrate various type of Caching (Data Caching, Output Caching, and SQL Server Cache Dependency).

10. Create an ASP .NET application using Ajax


12. Create a simple WCF application with appropriate client and host it in various environments IIS, Windows Service, Windows Application.

13. Create a Sequential Workflow.


15. Create a State Machine Workflow.


CA-Lab-VIII Lab on Linux Socket Programming

1. Implement TCP and UDP Client-Server programs for following services:
   Printing the Host ID, local port, also the client should indicate connection status
   Echo Service
   Day Time Service
   Chargen Service
   Mathematical Operation on numbers
   Checking number for prime, palindrome etc.
   Calculating factorial
   Calculating Fibonacci series
   Case conversion in given string

2. Implement Client-Server programs for demonstrating working of Concurrent Connection Oriented Servers using single process.

3. Implement Client-Server programs for demonstrating working of Concurrent Connection Oriented Servers using multiple processes.

4. Implement Telnet Server program for providing different types of Telnet Services.

5. Demonstrate and implement the file transfer using FTP.

6. Develop the Chat Client and Server program. The Server should be concurrent such as to provide intercommunication between multiple clients.

7. Develop a simple web server capable of accepting request from standard client like IE, Netscape, Opera etc (use standard protocol HTTP).
Semester- V

CA-501: Compiler Construction

Chapter – 1: Introduction to Compilation

Chapter - 2: Designing a Lexical Analyzer
Role of Lexical Analysis, Input Buffering, Specification of Tokens, Recognition of Tokens, Finite automata, Conversion from regular expression to NFA, Deterministic finite automata, Conversion from NFA to DFA, Minimization of DFA, Creating Lexical Analyzer with LEX.

Chapter – 3: Designing Syntax Analyzer
Role of Syntax Analyzer, Classification of parsers, Top-Down Parsing: Introduction, Problems in top-down parsing, Recursive Parsing, Problems in Recursive Procedures, Predictive Parsing,

**Chapter – 4: Intermediate Code Generation**


**Chapter – 5: Code Optimization**

Introduction, need for code optimization, Classification of code optimization techniques: Optimization techniques that work on machine code, Optimization techniques that work on intermediate forms of source code i.e. Optimization with in Basic Blocks: Folding, Redundant operation elimination, Optimization with in Loop: Strength Reduction, Dead code elimination, Moving operation within block out of block.

**Chapter – 6: Symbol Table Organization**

Introduction, Methods of organizing a symbol table: Unsorted, sorted symbol tables, binary search, hashing, its advantages, disadvantages, Collision, collision resolution techniques: Rehashing, Chaining.

**BOOKS:**


---

**CA-502 Software Engineering**

1. **Introduction:**
   What is Software Engineering. [2]

2. **Software Development Life-cycle:**
   Requirements analysis, software design, coding, testing, maintenance, etc. [6]

3. **Software Requirements Specification:**
   Waterfall model, prototyping, interactive enhancement, spiral model. Role of Management in software development. Role of metrics and measurement. [6]

4. **Software Requirement Specification:**
   Problem analysis, requirement specification, validation, metrics, monitoring and control. [6]
5. System Design: Problem partitioning, abstraction, top-down and bottom-up design, Structured approach. Functional versus object-oriented approach, design specification and verification metrics, monitoring and control.

6. Coding: Top-down and bottom-up, structured programming, information hiding, programming style, and internal documentation. Verification, Metrics, monitoring and control.

7. Testing: Levels of testing functional testing, structural testing, test plane, test cases specification, reliability assessment.

8. Software Project Management: Cost estimation, Project scheduling, Staffing, Software configuration management, Quality assurance, Project Monitoring, Risk management, etc.


BOOKS:


CA-503 Current Computing Trends-I

The syllabus of this course (CA-503: Current Computing Trends-I) will be prepared before the beginning of the Fifth Semester by the committee by considering the market requirement, student’s feedback and latest technological trends.

CA-504 Windows and Visual C++ Programming

1. Windows Programming:

2. Visual C++ Programming – Introduction:


3. The Document and View Architecture:


4. Active X and Object Linking and Embedding (OLE):

ActiveX controls Vs. Ordinary Windows Controls – Installing ActiveX controls – Calendar Control – ActiveX control container programming – create ActiveX control at runtime – Component Object Model (COM) – containment and aggregation Vs. inheritance – OLE drag and drop – OLE embedded component and containers – sample applications.

5. Advanced Concepts:


BOOKS:


CA-505 Current Computing Trends-II
The syllabus of this course (CA-505: Current Computing Trends-II) will be prepared before the beginning of the Fifth Semester by the committee by considering the market requirement, student's feedback and latest technological trends.

**CA-506 Soft Skills Development**

3. How to prepare a good Resume? [3]
4. Analytical/Logical Skill Development [8]
5. Interview Techniques [6]

*If the Resource Person is from the industry then he/she may decide the course content as the requirement.*

No grade will be given for the Soft Skills Development course. However, the completion of the credit for the CA-506 course shall be compulsory.

**BOOKS:**


**CA-Lab-IX Lab on Windows and Visual C++ Programming**

**Windows Programming:**

1. Write a window program to demonstrate line drawing with left mouse button. The color & width of the line should change with every new line
2. Write a Window Program that displays a small rectangle with every left mouse button. Double Clicking on existing rectangle should erase the rectangle.
3. Write a Window Program to display size of window and no. of left clicks; no. of right clicks and no. of double clicks. The data should be display at the center of the window. Size should be updated when user resizes the window object.
4. Write a Window Program to create filled rectangle and circle. on alternate left click. New figure should not erase the previous one. If user clicks inside any figure a mistake box is display.
5. Write a Window Program to create various brushes and change the background color to brushes on left mouse click.
6. Write a Window program to create a window object. Drag the left mouse buttons & display rectangle for which dragged line is a diagonal. Also demonstrate mouse capturing.
7. Write a window program to Move a ball horizontally inside the client area using timer. At each time lapse the ball should move left of the window and when it touches the left boundary of the window it should go to the right of the window and so on.
8. Write a window program to display the characters entered by user from the keyboard. [Consider only alphabets and numbers only.

MFC Programming:
1. Create a SDI application to display size of window & total number of Left clicks, right clicks and double clicks in the center of the client area.
2. Create a SDI application that displays a dialog box with two field’s viz. User name, Password and two push buttons. The dialog box is invoked as the user starts the application. After user pressed on ‘OIL’ button display the contents of username and password in the client area& the, password should be in the farm of sequences Of *
3. Create a SDI application, which invokes a dialog box from a menu option. The dialog box has three scroll bars called red scroll, blue scroll, green scroll and a multiline edit box. As the user scrolls through these scroll bars the background color of the edit box changes.
4. Create a SDI application that displays a dialog box with five fields: Name, Age (Spinner control.), Qualification it has three check boxes for MCA. M.Sc., M.Sc.-IT, Radio buttons for Sex & drop down combo box for Designation. As the user fills this information in this dialog box & pressed the ‘OK’ button. The information is display on the client area. The dialog should be invoked through a Menu option.
5. Create A SDI Application that invokes a dialog box from a menu called Dialog. The dialog has a track bar slider control, progress bar control and spinner control. As the slides through slider control the progress bar and spinner control should set the status. Display the value of current position set in trace bar.
6. Create a SDI application and implement modeless dialog box; data of one dialog box should be display in another dialog box when ok button of first dialog is press.
Note: - Both dialogs should be visible at same time.
7. Create a SDI application to create a dialog box with Multi column list control & display the student information in different columns. Student information: name, seat no, and class, rank should be considered.
8. Demonstrate splitter window, take Form view and simple view. The data entered in form should be displayed in the sample view.
9. Create an SDI application that contains an edit box; two buttons viz. Add, Remove & a list box. Whenever user enters a string in the edit box & presses the Add button data should be added to the List box & remove the data if:
i). It is present in the list.
ii). Press remove button at runtime.

10. Write a Program that capture Home, Page Up, Page Down, End & all arrows keys as user presses these keys. Program should display appropriate message in the client window.

11. Write a Window Program to draw sine wave.

12. Create a SDI application and create a dialog box with Single Column list box & a tree control. List contains family name and tree control contains family, hierarchy. As the user selects a family name from list a particular family in the tree control.

13. Create SDI application to Demonstrate Bitmaps. Also change the background color and icon of your application

14. Create a SDI application. Create a access database with a single table of your choice. Fill at least five six records in it. Display the contents of table in the multi column list view.

15. Create a SDI application with the following menu options. Display the selected menu item in the client area. Menu item should have check marks on it when selected.
   Cricket---------- Football
   One run -------- Corner
   Four run-------- Goal
   Sixer---------- Penalty Kick

CA-Lab-X Lab on Current Computing Trends

The syllabus of this Practical Course (CA LAB-X: Lab on Current Computing Trends) will be prepared before the beginning of the Fifth Semester by the Instructors based on the content of CA-503 and CA-505 Theory courses.

Semester- VI

CA-601 Full Time Industrial Training

Twelve credits shall be awarded to the Industrial Training/Project course, which will commence in VI Semester and the final work and report will be completed at the end of VI Semester for M.C.A. The marks and the credits will be allotted at the end of VI for M.C.A.