Course : S.Y. B. Sc. (Computer Science)

Proposed Syllabus (to be implemented from the academic year 2009-2010)

Pattern: Semester Pattern

Examination:

Pattern of Examination: Internal Examination (10 Marks) + University Examination (40 Marks) for each paper

Medium of Instruction: English

Equivalence of Subject: Table of Equivalance

Semester & Paper	**Title of Paper (Old Pattern) (Implemented from the academic year 2003-04)**	**Title of Paper (New Pattern) (to be implemented from the academic year 2009-10)**
Semester-I, Paper-I	CS-211, Data Structures, Image Structures and Related Algorithms in C	CS-211, Data Structures using C
Semester-I, Paper-II	CS- 212, File Structures and Database Concepts	CS-212, Relational Database Management System (RDBMS)
Semester-II, Paper-I	CS-221, Object Oriented Concepts and Programming in C++	CS-221, Object Oriented Concepts and Programming in C++
Semester-II, Paper-II	CS- 222, File Structures and Database Concepts	CS-222, Software Engineering

Theory Paper I
SEM I
DATA STRUCTURES USING C (CS 211)
(Compulsory Course)

Total Lectures: 48

Objective:

- To learn the systematic way of solving problem
- To understand the different methods of organizing large amount of data
- To efficiently implement the different data structures
- To efficiently implement solutions for specific problems

Prerequisites: Knowledge of C Programming Language

1. Introduction to data structures **[2]**

1.1 Concept
1.2 Data type, Data object, ADT
1.3 Need of Data Structure
1.4 Types of Data Structure

2. Algorithm analysis **[2]**

2.1 Algorithm – definition, characteristics
2.2 Space complexity, time complexity
2.3 Asymptotic notation (Big O, Omega Ω)

3. Linear data structures

3.1 Introduction to Arrays - array representation **[6]**
3.2 sorting algorithms with efficiency
- bubble sort, Insertion sort, Merge sort, Quick Sort

4. Linked List **[6]**

4.1 Introduction to List
4.2 Implementation of List – static & dynamic representation,
4.3 Types of Linked List
4.4 Operations on List
4.5 Applications of Linked List – polynomial manipulation
4.6 Generalized linked list – concept & representation

5. Stacks **[6]**

5.1 Introduction
5.2 Representation-static & dynamic
5.3 Operations
5.4 Application - infix to postfix & prefix, postfix evaluation,
5.5 Recursion using implicit stack
5.6 Concept of Multiple stacks

6. Queues **[8]**

6.1 Introduction
6.2 Representation -static & dynamic
6.3 Operations
6.4 Circuler queue, DeQue, priority queues
6.5 Concept of Multiple Queues

7. Trees **[12]**

7.1 Concept & Terminologies
7.2 Binary tree, binary search tree
7.3 Representation – static & dynamic
7.4 Operations on BST – create. Insert, delete, traversals (preorder, inorder, postorder), counting leaf, non-leaf & total nodes
7.5 Application - Heap sort
Height balance tree- AVL trees- Rotations

8. Graph **[6]**

8.1 Concept & terminologies
8.2 Graph Representation
8.3 Traversals – BFS & DFS
8.4 Applications – AOV network – topological sort
AOE network – critical path
Shortest path with implementation

References:

1. Fundamentals of Data Structures ---- By Horowitz Sahani (Galgotia)
2. Data Structures using C --- By ISRD Group (Tata McGraw Hill)
3. Introduction to Data Structures using C---By Ashok Kamthane
4. Data Structures using C --- Bandopadhyay & Dey (Pearson)

Theory Paper I
SEM II
Object Oriented Concepts and Programming in C++ (CS-221)
(Compulsory Course)

Total Lectures: 48
Objective:-

- Acquire an understanding of basic object oriented concepts and the issues involved in effective class design
- In order to write C++ programs that use object oriented concepts such as information hiding, constructors, destructors, inheritance etc.

Prerequisites: Knowledge of C Programming Language

1. Object oriented concepts **[2]**
- 1.1 Object oriented methodology
- 1.2 Features, advantages and Applications of OOPS

2. Introduction to C++ **[8]**
- 2.1 Data types, new operators and keywords, type conversion in C++
- 2.2 Introduction to reference variables
- 2.3 Classes & Objects
- 2.4 Classes & Object specifiers
- 2.5 Defining data members and member functions
- 2.6 Array of objects
- 2.7 Managing consol I/O
- 2.8 C++ stream classes
- 2.9 Formatted and unformatted console I/O
- 2.10 Usage of manipulators

3. Function in C++ **[6]**
- 3.1 Call by reference, Return by reference
- 3.2 Function overloading and default arguments
- 3.3 Inline function
- 3.4 Static class members
- 3.5 Friend functions

4. Constructors and destructor **[4]**
- 4.1 types of constructors
- 4.2 memory allocation (new and delete)
- 4.3 usage of destructor

5. Operator overloading **[4]**
- 5.1 overloading unary and binary operators
- 5.2 overloading using friend function
- 5.3 usage of this pointer
- 5.4 overloading insertion and extraction operator

6. Inheritance **[10]**
- 6.1 types of inheritance with examples
- 6.2 virtual base classes and abstract base classes
- 6.3 constructor and destructor in derived class
- 6.4 virtual functions and pure virtual function

7. Working with files **[6]**
- 7.1 File operations
- 7.2 File pointer and their manipulation
- 7.3 File updation with random access

8. Templates **[4]**
- 8.1 Introduction to templates,
- 8.2 Class templates, function templates and overloading of function templates
- 8.3 With multiple parameters
- 8.4 CASE study on STL (with reference to container classes, operational utilities)

9. Exception Handling in C++ **[4]**

9.1 try, catch and throw primitives

Reference Books: -

1. Object Oriented Programming with C++ by Robert Lafore
2. Object Oriented Programming with C++ by E. Balagurusamy
3. Object Oriented Modeling and Design by James Rambough
4. The Complete Reference C++ by Herbert Shildth
5. Let us C++ by – Yashwant Kanitkar

Theory Paper – II
SEM I
Relational Database Management System (RDBMS)(CS-212)
(Compulsory Course)

Total Lectures: 48

Objective:-

- To teach fundamental concepts of RDBMS (MySQL)
- To teach principles of databases
- To teach database management operations
- To teach data security and its importance
- To teach client server architecture

Prerequisites: Knowledge of DBMS

1. **MySQL** **[12]**

1.1. Creating a Database and Tables
1.2. Inserting, Selecting, Ordering, Limiting, Grouping, Analyzing and Manipulating Data
1.3. Changing, Deleting, Searching, Importing Data
1.4. Command Line Interface
1.5. Database and Table Schema Statements
1.6. Data Manipulation Statements and Functions
1.7. Table Statements and Functions
1.8. Replication Statements and Functions
1.9. Stored Routine Statements
1.10 Aggregate Clauses, Aggregate Functions
1.11 String Functions
1.12 Date and Time Functions
1.13 Mathematical Functions
1.14 Flow Control Functions
1.15 Stored Functions and Cursors
1.16 Stored Procedures, Views and Triggers
1.17 Exception Handling

2 **Transaction Concepts** **[14]**

2.10 Describe a transaction, properties of transaction, state of the transaction.
2.11 Executing transactions concurrently associated problem in concurrent execution.
2.12 Schedules, types of schedules, concept of serializability, precedence graph for Serializability.
2.13 Ensuring Serializability by locks, different lock modes, 2PL and its variations.
2.14 Basic timestamp method for concurrency, Thomas Write Rule.
2.15 Locks with multiple granularity, dynamic database concurrency (Phantom Problem).
2.16 Timestamps versus locking.
2.17 Deadlock handling methods
2.18 Detection and Recovery (Wait for graph).
2.19 Prevention algorithms (Wound-wait, Wait-die)

3 **Database Security Concepts** **[8]**

3.10 Introduction to database security concepts
3.11 Methods for database security
3.12 Discretionary access control method
3.13 Mandatory access control and role base access control for multilevel security.
3.14 Use of views in security enforcement.
3.15 Overview of encryption technique for security.
3.16 Statistical db security.

4 **Crash Recovery** **[8]**

4.1 Failure classification
4.2 Recovery concepts
4.3 Log base recovery techniques (Deferred and Immediate update)
4.4 Checkpoints
4.5 Recovery with concurrent transactions (Rollback, checkpoints, commit)
4.6 Database backup and recovery from catastrophic failure.

5. **Client-Server Technology** **[6]**
    5.1 Describe client-server computing.
    5.2 Evolution of Client - Server information systems.
    5.3 Client – Server Architecture benefits.
    5.4 Client Server Architecture
        - Components, Principles, Client Components
        - Communication middleware components
        - Database middleware components
        - Client Server Databases

References:-

1. Fundamentals of Database Systems (4th Ed) By: Elmasri and Navathe
2. Database System Concepts (4th Ed) By: Korth, Sudarshan, Silberschatz
3. MySQL The Complete Reference By Vikram Vaswani
4. Learning MySQL by O'reilly
5. MySQL in Nut Shell by Dyer 2nd Edition

Theory Paper – II
SEM II
Software Engineering (CS 222)
(Compulsory Course)

Total Lectures: 48
Objective:-

- To teach concepts of Software Engineering
- To teach principles of Software Engineering
- To teach various process models used in practice
- To know about the system engineering and requirement engineering
- To build analysis model

Prerequisites: Basic knowledge of system concepts and DBMS

1. Introduction To Software Engineering **[4]**
- 1.1 The Evolving Role of Software
- 1.2 Software
- 1.3 The Changing Nature of Software
- 1.4 Legacy Software
  - 1.4.1 The Quality of Legacy Software
  - 1.4.2 Software Evolution
- 1.5 Software Myths

2. A Generic View of Process **[6]**
- 2.1 Software Engineering – A Layered Technology
- 2.2 A Process Framework
- 2.3 Personal and Team Process Models
  - 2.3.1 Personal Software Process (PSP)
  - 2.3.2 Team Software Process (TSP)
- 2.4 Process Technology
- 2.5 Product and Process

3. Process Models **[6]**
- 3.1 Prescriptive Models
- 3.2 The Waterfall Model
- 3.3 Incremental Process Models
  - 3.3.1 The Incremental Model
  - 3.3.2 The RAD Model
- 3.4 Evolutionary Process Models
  - 3.4.1 Prototyping
  - 3.4.2 The Spiral Model
  - 3.4.3 The Concurrent Development Model
  - 3.4.4 A Final Comment of Evolutionary Processes

4. An Agile View of Process **[4]**
- 4.1 What Is Agility?
- 4.2 What Is an Agile Process?
  - 4.2.1 The Politics of Agile Development
  - 4.2.2 Human Factors
- 4.3 Agile Process Models
  - 4.3.1 Extreme Programming (XP)
  - 4.3.2 Adaptive Software Development (ASD)
  - 4.3.3 Dynamic Systems Development Method (DSDM)
  - 4.3.4 Scrum
  - 4.3.5 Crystal
  - 4.3.6 Feature Driven Development (FDD)
  - 4.3.7 Agile Modeling (AM)

5. Software Engineering Practice **[6]**

Software Engineering Practice
- The Essence of Practice
- Core Principles

Communication Practices

Planning Practices

Modeling Practices
- Analysis Modeling Principles
- Design Modeling Principles

6. System Engineering **[4]**

6.1 Computer-Based Systems
6.2 The System Engineering Hierarchy
- 6.2.1 System Modeling
- 6.2.2 System Simulation

6.3 Business Process Engineering: An Overview

7. Requirements Engineering **[10]**

7.1 A Bridge to Design and Construction
7.2 Requirements Engineering Tasks
- 7.2.1 Inception
- 7.2.2 Elicitation
- 7.2.3 Elaboration
- 7.2.4 Negotiation
- 7.2.5 Specification
- 7.2.6 Validation
- 7.2.7 Requirements Management

7.3 Initiating the Requirements Engineering Process
- 7.3.1 Identifying the Stakeholders
- 7.3.2 Recognizing Multiple Viewpoints
- 7.3.3 Working Toward Collaboration
- 7.3.4 Asking the First Questions

7.4 Eliciting Requirements
- 4.4.1 Collaborative Requirements Gathering
- 4.4.2 Quality Function Deployment
- 4.4.3 User Scenarios
- 4.4.4 Elicitation Work Products

7.5 Building the Analysis Model
- 7.5.1 Elements of the Analysis Model
- 7.5.2 Analysis Patterns

7.6 Negotiating Requirements
7.7 Validating Requirements

8. Building the Analysis Model **[8]**

8.1 Requirements Analysis
- 8.1.1 Overall objective and Philosophy
- 8.1.2 Analysis rule of Thumb
- 8.1.3 Domain Analysis

8.2 Analysis Modeling Approaches
8.3 Data Modeling Concepts
- 8.3.1 Data Objects
- 8.3.2 Data Attributes
- 8.3.3 Relationships
- 8.3.4 Cardinality and Modality

Reference Books:

1. Software Engineering – A Practitioner's Approach 7th Edition – Roger S. Pressman [McGraw Hill International Edition]
2. Software Engineering – IAN Sommerville 7th / 8th Edition (Pearson Edition)

S.Y.B.Sc. (Computer Science)
Lab Course-I
Practical Assignment

Teaching Scheme:- 4 Lectures Per Week per batch of 15 students

Examination Scheme:-

Practical examination will be conducted by respective colleges at the end of academic year, 80 marks will be assigned to practicals and 20 marks for journals and orals.

LAB I: Data Structures using C Assignments

1. Sorting Algorithms – Bubble sort, Insertion, selection, quick sort and merge.
2. Static/Dynamic stack implementation, infix to postfix, infix to prefix and evaluation of Postfix.
3. Static and Dynamic Queue Implementation.
4. Singly Linked List, Doubly Linked List and Circular Linked List.
5. Polynomial addition (Using Linked list).
6. Binary Tree Traversal: Create, add, delete, display nodes.
7. Graph: in degree, out degree, DFS, BFS.
8. Shortest path Dijkstra algorithm.
9. Adjacency matrix to adjacency list conversion.

LAB I: C++ Assignments

1. Class and Object, Array of Objects.
2. Inline function, friend function, default argument, function overloading.
3. Operator Overloading
4. Constructor: Copy Constructor, Default Constructor, Parameterized Constructor.
5. Memory Allocation: new and delete operators
6. Inheritance: Single, multiple, multilevel, hierarchy.
7. File Handling: Updation of files using random access
8. Templates: Function and Class.

S.Y.B.Sc. (Computer Science)
Lab Course- II
Practical Assignment

Teaching Scheme:- 4 Lectures Per Week per batch of 15 students

Examination Scheme:-

Practical examination will be conducted by respective colleges at the end of academic year. There will be grading system based on performance of candidates.

The grading system is as follows.

Marks	**Grade**	
Below 40	D	(D Grade indicates fail)
40 - 49	C	
50 - 59	C+	
60 - 69	B	
70 - 79	B+	
80 - 89	A	
90 and Above	A+	

Q. 1. My-SQL Assignments

- Simple and Nested Queries
-Queries using function
-Queries using cursors
-Stored procedure and function
-triggers
-cursor and database schema

Q. 2. Mini Project based on SE Concept

- Problem Definition
- Scope of the system
- Proposed System
-Fact finding techniques
-Feasibility study
-ERD
-DFD
-I/O Screens
-O/P Formats
-Report Layout
-Conclusion
-Bibliography

NOTE: A booklet should be made available to the students for the Lab Course I and II assignments.