OBJECT ORIENTED SYSTEMS
(CS301)

Objective: The objective of the course is to acquaint the student of object - oriented systems.	
Unit	**Topic**
I	**Object Modeling:** Objects and classes, links and association, generalization and inheritance, aggregation, abstract class, multiple inheritance, meta data, candidate keys, constraints.
II	**Dynamic Modeling:** Events and states, operations, nested state diagrams and concurrency, advanced dynamic modeling concepts, a sample dynamic model.
III	**Functional Modeling:** Data flow diagram, specifying operations, constraints, a sample functional model.OMT (object modeling techniques) methodologies, examples and case studies to demonstrate methodologies, comparisons of methodologies, SA/SD, JSD.
IV	Introduction of Classes, Specifying a Class, Defining a Member Functions, A C++ Program with Class Access Specifiers, Inline functions, Nesting of Member Functions, Memory Allocation for Objects, Static Data Members, Static Member Functions, Arrays of Objects, Objects as Function Arguments, Default Arguments, Const Arguments, Function Overloading, Friend Functions.
V	Introduction, Constructors, Parameterized Constructors, Multiple Constructors in a Class, Constructors with Default Arguments, Dynamic initialization of Objects, Copy Constructors, Dynamic Constructors, Destructors. Introduction to inheritance, Defining Derived Classes, Single Inheritance, Multiple Inheritance, Multi Level Inheritance, Hierarchical Inheritance, Hybrid Inheritance, Abstract Classes, Constructors in Derived Classes, Containership, Operator overloading, Rules for Operator overloading, overloading of binary and unary operators.

References:

1. Object Oriented Modeling and Design by Et Al Rumbaugh and Blaha, Pearson, 2007.
2. Object Oriented Programming in C++ by E.Balagurusamy., published by TMH, 2013.
3. Design Patterns - Elements of Reusable Object-Oriented Software, Gamma, , Addison-Wesley, 1994.
4. Mastering C++ by K.R.Venugopal., published by Tata McGraw- Hill, 2006.

DATABASE MANAGEMENT SYSTEM
(CS302)

Objective: To educate students with fundamental concepts of Database Management System, Data Models, Different Data Base Languages and how they differ from traditional file system.

Unit	Topic
I	**Introduction:** An overview of database management system, database system vs. file system, Database system concept and architecture, data model schema and instances, data independence and database language and interfaces, data definitions language, DML, Overall Database Structure. **Data Modeling using the Entity Relationship Model:** ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation, reduction of an ER diagrams to tables, extended ER model, relationship of higher degree.
II	**Relational data Model and Language:** Relational data model concepts, integrity constraints, entity integrity, referential integrity, Keys constraints, Domain constraints, relational algebra, relational calculus, and tuple and domain calculus. **Introduction on SQL:** Characteristics of SQL, advantage of SQL. SQL data type and literals, Types of SQL commands, SQL operators and their procedure, Tables, views and indexes, Queries and sub queries, Aggregate functions. Insert, update and delete operations, Joins, Unions, Intersection, Minus, Cursors, Triggers and Procedures in SQL/PLSQL.
III	**Data Base Design & Normalization:** Functional dependencies, normal forms, first, second, third normal forms, BCNF, inclusion dependence, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design.
IV	**Transaction Processing Concept:** Transaction system, Testing of serializability, serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log based recovery, checkpoints, deadlock handling.
V	**Distributed Database:** distributed data storage, Concurrency Control Techniques: Concurrency control, Locking Techniques for concurrency control, Time stamping protocols for concurrency control, validation based protocol, multiple granularity, Multi version schemes, Recovery with concurrent transaction, case study of Oracle.

References:

1. Date C.J., "An Introduction to Database System", Addison Wesley, 1985.
2. Korth, Silbertz, Sudarshan, "Database Concepts", Mc Graw Hill, 2006.
3. Elmasri, Navathe, "Fundamentals of Database System", Addison Wesley, 2016.
4. Bipin C. Desai, "An Introduction to Database System", Galgotia Publication, 2006.
5. Majumdar & Bhattacharya, " Database Management System", TMH, 2004.

DATA STRUCTURES
(CS303)

Objective: To discuss common sorting, searching and graph algorithms, and to study the complexity and comparisons among these various techniques.	
Unit	**Topic**
I	**Introduction:** Basic Terminology, Elementary Data Organization, Data Structures, Operations on data structures, Algorithm, Complexity. **Arrays:** Definition, Single and Multidimensional Arrays, Representation of Arrays, Operations on an array, Application of arrays, Sparse Matrices and their representations. **Sorting and Searching:** Insertion Sort, Selection Sort, Bubble Sort, Merge Sort, Radix Sort, Linear Search, Binary Search, Hash Function, Collision Resolution Strategies.
II	**Pointers:** Definition and declaration, Pointer variables, Pointer arithmetic, Pointer Arrays, Pointer to structure, Dynamic Memory Allocation, Garbage collection. **Linked lists:** Array Implementation and Dynamic Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List; Operations on a Linked List: Insertion, Deletion, Traversal; Polynomial Representation and Addition, Generalized Linked List.
III	**Stacks:** Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Recursion, Tower of Hanoi Problem, Tail recursion, Removal of recursion. **Queues:** Operations on Queue: Create, Add, Delete, Full and Empty; Array and linked implementation of queues in C, Circular queues, Dequeue and Priority Queue.
IV	**Trees**: Basic terminology, Binary Trees, Complete Binary Tree, Extended Binary Trees, Algebraic Expressions, Array and Linked Representation of Binary trees, Tree Traversal algorithms: Inorder, Preorder and Postorder, Binary Search Trees (BST), Insertion and Deletion in BST, Threaded Binary trees, Traversing in Threaded Binary Trees, AVL trees, Heap Creation, Heap Sort, Huffman algorithm, General Tree, B Trees & B+ Trees.
V	**Graphs:** Terminology, Sequential and linked Representations of Graphs: Adjacency Matrices, Adjacency List, Adjacency Multi list; Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Shortest Path algorithm: Warshal Algorithm and Dijikstra Algorithm.

References:

1. Data Structures With C by Seymour Lipschutz, TMH, 2014.
2. Data Structures by Horowitz and Sahani, Computer Science Press, 1998.
3. Data Structures Using C And C++ By Tanenbaum, PHI, 1998.

DISCRETE STRUCTURES AND GRAPH THEORY
(CS304)

Objective: Introduce a number of Discrete Mathematical Structures (DMS) found to be serving as tools even today in the development of theoretical computer science.

Unit	Topic
I	**Set Theory:** Introduction, Combination of sets, Multisets, Ordered pairs, Proofs of some general identities on sets. **Relations:** Definition, Operations on relations, Properties of relations, Composite Relations, Equality of relations, Recursive definition of relation, Order of relations. **Functions:** Definition, Classification of functions, Operations on functions, Recursively defined functions, Growth of Functions. **Natural Numbers:** Introduction, Mathematical Induction, Variants of Induction, Induction with Nonzero Base cases. Proof Methods, Proof by counter – example, Proof by contradiction.
II	**Algebraic Structures:** Definition, Groups, Subgroups and order, Cyclic Groups, Cosets**,** Lagrange's theorem, Normal Subgroups, Permutation and Symmetric groups, Group Homomorphism's, Definition and elementary properties of Rings and Fields, Integers-Modulo-n.
III	**Partial order sets:** Definition, Partial order sets, Combination of partial order sets, Hasse-diagram. **Lattices:** Definition, Properties of lattices – Bounded, Complemented, Modular and Complete lattice. **Boolean algebra:** Introduction, Axioms and Theorems of Boolean algebra, Algebraic manipulation of Boolean expressions, Simplification of Boolean Functions, Karnaugh-maps, Logic gates, Digital circuits and Boolean algebra.
IV	**Propositional Logic:** Proposition, well-formed formula, Truth tables, Tautology, Satisfiability, Contradiction, Algebra of proposition, Theory of Inference, Natural Deduction. **Predicate Logic:** First order predicate, well-formed formula of predicate, quantifiers, Inference theory of predicate logic.
V	**Trees:** Definition, Binary tree, Binary tree traversal, Binary search tree. **Graphs:** Definition and terminology, Representation of graphs, Multi-graphs, Bipartite graphs, Planar graphs, Isomorphism and Homeomorphism of graphs, Euler and Hamiltonian paths, Graph coloring. **Recurrence Relation & Generating function:** Recursive definition of functions, Recursive algorithms, Method of solving recurrences. **Combinatory:** Introduction, Counting Techniques, Pigeonhole Principle.

References:

1. Trembley, J.P & R Manobar, "Discrete Mathematical Structure with Application to Computer Science", TMH, 2014.
2. Kenneth H. Rosen, "Discrete Mathematics and its Application", TMH, 2016.
3. Mathematical Structure for Computer Science, Gersting, WH Freeman & Macmillam, 2014.

OBJECT ORIENTED SYSTEMS LAB
(CS351)

LIST OF EXPERIMENTS

1. Design C++ classes with static members, methods with default arguments.
2. Implement complex number class with necessary operator overloading and type conversions such as integer to complex, double to complex, complex to double etc using C++.
3. Implement Matrix class with dynamic memory allocation and necessary methods. Give proper constructor, destructor, copy constructor, and overloading of assignment operator.
4. Manage bank account using inheritance concept using C++
5. Design stack and queue classes with necessary exception handling using C++.
6. Develop with suitable hierarchy, classes for Point, Shape, Rectangle, Square, Circle, Ellipse, Triangle, Polygon, etc. Design a simple test application to demonstrate dynamic polymorphism and RTTI using C++.

DATABASE MANAGEMENT SYSTEM LAB
(CS352)

LIST OF EXPERIMENTS

1. Write the queries for Data Definition and Data Manipulation Language.
2. Write SQL queries using logical operators.
3. Write SQL queries using SQL operators
4. Write SQL query using character, number, date and group functions
5. Write SQL queries for relational algebra
6. Write SQL queries for extracting data from more than one table
7. Write SQL queries for sub queries, nested queries
8. Write program of PL/SQL
9. Concepts for ROLL BACK, COMMIT & CHECK POINTS
10. Create VIEWS, CURSORS and TRGGERS

DATA STRUCTURES LAB
(CS353)

LIST OF EXPERIMENTS

1. To implement addition and multiplication of two 2D arrays.
2. To transpose a 2D array.
3. To implement stack using array.
4. To implement queue using array.
5. To implement circular queue using array.
6. To implement stack using linked list.
7. To implement queue using linked list.
8. To implement circular queue using linked list.
9. To implement binary tree using linked list.
10. To implement binary search tree using linked list.
11. To implement tree traversals using linked list.
12. To implement BFS using linked list.
13. To implement DFS using linked list.
14. To implement Linear Search.
15. To implement Binary Search.
16. To implement Bubble Sorting.
17. To implement Selection Sorting.
18. To implement Insertion Sorting.
19. To implement Merge Sorting.
20. To implement Heap Sorting.

OPERATING SYSTEM
(CS401)

Objective: To give knowledge about principles of modern operating system.	
Unit	**Topic**
I	**Introduction to Operating Systems:** Role and purpose of operating systems, Operating System Services, Classification of Operating systems, Operating System Structure, System Calls.
II	**CPU Scheduling:** Process vs. Program, Process States, Process Transition Diagram, Process Control Block, Process Address Space, Schedulers, Scheduling Concepts, Performance Criteria, Scheduling Algorithms, Threads, Deadlock Problem, Deadlock Characterization, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock, Methods for Deadlock Handling.
III	**Inter-process Communication:** Race conditions, Critical sections, Mutual exclusion, Critical-section problem, Algorithmic approach to implementing critical sections, Hardware support for process synchronization, Semaphores, Mutexes, Monitors. **Classic Problems of Synchronization:** Producers-consumers with bounded buffers problem, Readers-writers problem, Dining-philosophers problem
IV	**Memory Management:** Introduction, Logical and Physical Address Space, Swapping, Contiguous Memory Allocation, Fragmentation, Paging, Structure of Page Table, Segmentation, Segmentation with Paging. Virtual Memory: Demand Paging, Performance of Demand Paging, Page Replacement Algorithms, Allocation of Frames, Thrashing
V	**File system:** File Concept, Access Methods, Directories, Mounting of File-System, File-System Structure, File-System Implementation, Allocation Methods. I/O Devices, and I/O Subsystems, I/O Buffering, Disk Storage and Disk Scheduling, Disk Management, RAID.

References:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts, John Wiley, 2008,
2. William Stallings, Operating Systems: Internals and Design Principles. PHI, 2012.
3. AS Tanenbaum, Modern Operating Systems, Pearson, 2015.
4. AS Tanenbaum, AS Woodhull, Operating Systems Design and Implementation, PHI, 2006.

COMPUTER GRAPHICS
(CS402)

Objective: The goal of subject is to provide the students a broad exposure to the computer graphics field in order to prepare for follow-on study.	
Unit	**Topic**
I	**Introduction and Line Generation:** Types of computer graphics, Graphic Displays Random scan displays, Raster scan displays, Frame buffer and video controller, Points and lines, Line drawing algorithms, Circle generating algorithms, Mid-point circle generating algorithm, and parallel version of these algorithms
II	**Transformations:** Basic transformation, Matrix representations and homogenous coordinates, Composite transformations, Reflections and shearing. **Windowing and Clipping:** Viewing pipeline, Viewing transformations, 2-D Clipping algorithms- Line clipping algorithms such as Cohen Sutherland line clipping algorithm, Liang Barsky algorithm, Line clipping against non rectangular clip windows; Polygon clipping – Sutherland Hodgeman polygon clipping, Weiler and Atherton polygon clipping,Curve clipping, Text clipping
III	**Three Dimensional:** 3-D Geometric Primitives, 3-D Object representation, 3-D Transformation, 3-D viewing, projections, 3-D Clipping.
IV	**Curves and Surfaces**: Quadric surfaces, Spheres, Ellipsoid, Blobby objects, Introductory concepts of Spline, Bspline and Bezier curves and surfaces.
V	**Hidden Lines and Surfaces:** Back Face Detection algorithm, Depth buffer method, Abuffer method, Scan line method, basic illumination models– Ambient light, Diffuse reflection, Specular reflection and Phong model, Combined approach, Warn model, Intensity Attenuation, Color consideration, Transparency and Shadows.

References:

1. Donald Hearn and M. Pauline Baker, " Computer Graphics", PHI, 2016.
2. Steven Harrington, "Computer Graphics: A Programming Approach", TMH, 1987.
3. Foley James D, "Computer Graphics", AW, 2014

WEB TECHNOLOGY
(CS403)

Objective: To provide a basic understanding about web world and develop web sites based projects.	
Unit	**Topic**
I	**Introduction to Web Technology**: Internet, WWW, Web Browsers with suitable examples, Web Servers with suitable examples, URL, HTTP, MIME. Introduction to HTML& DHTML : Basic Syntax, HTML Document Structure , Text Formatting, Images, Lists, Links, Tables, Frames, Forms. Cascade Style Sheets**:** Levels Of Style Sheets, Specification Formats, Style Classes, Properties, Colors, Span and Div tags.
II	**Introduction to Java Script:** Overview of java Script, Syntactic characteristics, Primitives, Operator and Expression, control statements, Arrays, functions, errors in scripts, Document Object Model(DOM),Event driven computation, element access in Java script, The navigator Object. Dynamic Document with Java Script : Element positioning, Moving elements, Changing colors and fonts, Dynamic content, Locating the mouse Cursor, Slow movements of elements, Dragging and Dropping Elements.
III	**Introduction to XML:** Syntax of XML, Document Structure, Document type definition, Namespaces,XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX.
IV	**Introduction to PHP:** Overview of PHP, general server characteristics, Creating PHP Pages, Form handling , Data Base access with PHP & MySql. JSP & Servlets: Basics of JSP, Tags, Session Handling, Redirection and Basics of Servlet
V	**Database Connectivity:** Connection to various databases (MySQL, Oracle). Database connectivity in PHP, JSP, Other server languages, Basic SQL Queries and Statements.

References:

1. Programming world wide web- Robert W.Sebesta , Pearson, 2008.
2. Beginners PHP, Apache, MY Sql, Web Development, by Michael Glass, Wrox, 2004.

COMPUTER ORGANIZATION &ARCHITECTURE
(CS404)

Unit	Topic
I	**Functional blocks of a computer**: CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU–registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study – instruction sets of some common CPUs. **Data representation**: signed number representation, fixed and floating point representations, character representation. Computer arithmetic – integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-and add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic.
II	**Introduction** to x86 architecture. **CPU control unit design**: hardwired and micro-programmed design approaches, Case study – design of a simple hypothetical CPU. **Memory system design**: semiconductor memory technologies, memory organization.
III	**Peripheral devices and their characteristics**: Input-output subsystems, I/O device interface, I/O transfers–program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes‑role of interrupts in process state transitions, I/O device interfaces ‑ SCII, USB
IV	**Pipelining**: Basic concepts of pipelining, throughput and speedup, pipeline hazards. **Parallel Processors**: Introduction to parallel processors, Concurrent access to memory and cache coherency.
V	**Memory organization:** Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.

Text Book (s):

1. “Computer Organization & Architecture”, Rajaraman, PHI Learning
2. “Computer Organization and Embedded Systems”, 6th Edition by Carl Hamacher, McGraw Hill Higher Education.

OPERATING SYSTEM LAB
(CS451)

LIST OF EXPERIMENTS

1. Write a program to implement fork system call.
2. Write a program to implement critical section problems.
3. Write a program to implement classical problems of process synchronization.
4. Write a program to implement non-preemptive scheduling algorithm
5. Write a Program to implement preemptive scheduling algorithm.
6. Write a program to Implement Banker's algorithm.
7. Write a program to implement page replacement algorithms.
8. Write a program to implement Disk Scheduling algorithms.
9. Write a Program to implement file allocation methods.
10. Write a Program for MFT and MVT first fit and best fit.

COMPUTER GRAPHICS LAB
(CS452)

LIST OF EXPERIMENTS

1. Write a program to animate a bouncing ball using in built functions.
2. Write programs to draw Line, circle using DDA technique and polynomial equation.
3. Write programs to implement Mid point line, circle.
4. Write programs to implement Bresenham's line, circle.
5. Write programs to implement Flood-fill , boundary fill algorithm.
6. Write a program to implement scan line polygon fill algorithm.
7. Write a program to implement all the basic 2D-Transformations.
8. Write a program to implement cohen-sutherland line clipping algorithm.
9. Write a Program to implement sutherland-Hodgmn clipping algorithm.
10. Write a Program to implement Bezier curve.

WEB TECHNOLOGY LAB
(CS453)

LIST OF EXPERIMENTS

1. Write a program in HTML to display different styles of heading text.
2. Write a program to display the processes to be followed for a patient when he enters for a complete check-up. Use ordered lists and unordered lists.
3. Write a program to display a traditional Newspaper with the use of table tags.
4. With the help of "IMAGE" tags write a program to display the image along with some contents.
5. Use "Anchor" tag to write a program for displaying various Menus.
6. Use mapping technique, to map a particular part of image and move the control corresponding to that area. For e.g., in an image, if there are bat, ball, stump etc. When you click stump control should move to a file call St.htm.
7. Create frames that have details about various cities.
8. Create a form to display the kinds of food available in a Restaurant. (Use checkboxes wherever necessary)
9. Write a program to "reload" a page automatically once in 5 seconds.
10. Write a program using CSS to set the background colour, font, and paragraph.

COMPUTER ORGANIZATION &ARCHITECTURE LAB

(CS454)

LIST OF EXPERIMENTS

1. Implementing HALF ADDER, FULL ADDER using basic logic gates
2. Implementing Binary -to -Gray, Gray -to -Binary code conversions.
3. Implementing 3-8 line DECODER.
4. Implementing 4x1 and 8x1 MULTIPLEXERS.
5. Verify the excitation tables of various FLIP-FLOPS.
6. Design of an 8-bit Input/ Output system with four 8-bit Internal Registers.
7. Design of an 8-bit ARITHMETIC LOGIC UNIT.
8. Design the data path of a computer from its register transfer language description.
9. Design the control unit of a computer using either hardwiring or microprogramming based on its register transfer language description.
10. Implement a simple instruction set computer with a control unit and a data path.