ANDHRA UNIVERSITY COLLEGE OF ENGINEERING (A): VISAKHAPATNAM

Branch: INFORMATION TECHNOLOGY

II/IV B.TECH (IT) (FOUR YEAR COURSE)

(With effect from 2019-2020 admitted batch onwards)

B.TECH (IT) 2nd YEAR I-SEMESTER SCHEME OF INSTRUCTION AND EXAMINATION
With effect from 2019-2020 admitted batch

Sub Ref. No	Name of the Subject	Periods			Maximum marks			Credits
		Theory	Tutorial	Lab	Exam	Internal	Total	
CSE211	ELEMENTS OF ELECTRONICS ENGINEERING	3	0	--	70	30	100	3
CSE212	DATA STRUCTURES & ALGORITHMS	3	1	--	70	30	100	4
CSE213	DIGITAL LOGIC DESIGN	3	0	--	70	30	100	3
CSE214	OBJECT ORIENTED PROGRAMMING	3	1	--	70	30	100	4
IT215	SYSTEMS PROGRAMMING	3	0	--	70	30	100	3
CSE216	PRINCIPLES OF ECONOMICS AND MANAGEMENT	3	0	--	70	30	100	3
CSE217	DATA STRUCTURES LAB	--	--	3	50	50	100	1.5
CSE218	OBJECT ORIENTED PROGRAMMING LAB	--	--	3	50	50	100	1.5
	TOTAL	18	2	6	520	280	800	**23**

CSE 211	**ELEMENTS OF ELECTRONICS ENGINEERING** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week,	Univ. Exam: 3 Hours	**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction to Electronics and Semiconductors:** Energy band theory, Conduction in Insulators, Semiconductors and metals, Electron emission from metals, Classification of semiconductors, Carrier concentration in an intrinsic semiconductor, Properties of intrinsic semiconductor, Drift and diffusion currents.
2. **Semi-Conductor Diode:** Theory of PN junction diode, Open circuited PN junction, V-I characteristics of a PN diode, Diode current equation, Transition and diffusion capacitances, Break down in PN diode, Applications of PN diodes. Zener diode, Zener regulator, Tunnel diode, Schottky diode.
3. **Rectifying circuits:** Half wave and full wave rectifiers, Bridge rectifiers, Efficiency, Ripple and regulation of each rectifier, Capacitor filters.
4. **Bipolar Junction Transistor** :– Introduction, construction, Operation of PNP and NPN Transistors – Transistor Circuit configurations- Characteristics of a CE configurations – h parameters, low frequency small signal equivalent circuit of a Transistor.
5. **Transistor Biasing and thermal stabilization:** Transistor Biasing, Stabilization, Different methods of transistor biasing – Fixed bias, Collector feedback bias – self bias – Bias compensation.
6. **Transistor Amplifiers:** CE, CB, CC amplifier configurations –Multistage amplifier – A Two Stage RC coupled amplifier – frequency response curve and bandwidth**.**
7. **Field Effect Transistors:** Junction Field Effect Transistors (JFET) – JFET characteristics, JFET Parameters, Small signal equivalent circuit – MOSFETS – Depletion and Enhancement MOSFETS.

Text Books:

1. Electronic Device and Circuits by Sanjeev Guptha.

Reference Books:

1. Electronic Device and Circuits Theory by Robert L. Boylested Electronic Device and Circuits by David. A. Bell

CSE 212	DATA STRUCTURES & ALGORITHMS *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods+1Tut/week, Univ. Exam: 3 Hours		**Credits: 4**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction to Data Structures**: Review of C Programming, Recursive Definition and Processes, Recursion in C, Simulation of Recursion, Efficiency of Recursion, Abstract Data Types, Meaning and Definition of Data Structures, Arrays.
2. **Stacks**: Stack as an Abstract Data Type, Primitive Operations, Implementing Stack Operations using Arrays, Infix, Postfix and Prefix: Definitions, Evaluation and Conversions. **Queues**: Queue as an Abstract Data Type, Sequential Representation, Types of Queues, Operations, Implementation using Arrays.
3. **Linked List**: Operations, Implementation of Stacks, Queues and priority Queues using Linked Lists+, Circular Lists: Insertion, Deletion and Concatenation Operations, Stacks and Queues as Circular Lists, Doubly Linked Lists.
4. **Trees**: Binary Trees - Definitions and Operations, Binary Tree Representation: Node Representation, Implicit array Representation, Binary Tree Traversal, Threaded Binary Trees and their Traversal, Trees and their Applications; Heterogeneous binary trees.
5. **Searching**: Basic Searching Techniques: Dictionary as an Abstract Data Type, Algorithmic Notation, Sequential Searching and its Efficiency, Binary Search, Interpolation Search. Tree Searching: Insertion and Deletion of a node from a Binary Search Tree, Efficiency of Binary Search Tree operations.
6. **Sorting:** General Background: Efficiency, Asymptotic Notations, Efficiency of Sorting, Bubble Sort and Quick Sort and their Efficiency, Selection Sorting, Binary Tree Sort, Heap Sort, Insertion Sorts, Shell Sort, Address calculation Sort, Merge and Radix Sorts.
7. **Graphs and Their Application:** Definition of Graphs, Representation of Graphs, Transitive closure, Linked Representation of Graphs, Topological Ordering of nodes, Graph Traversal and Spanning Forests, Undirected Graphs and their Traversals, Applications of Graphs, Minimal Spanning Trees.

Text Books:

1. Data Structures Using C and C++ Yaddish Langsam, MosheJ. Augenstein and Aaron M. Tanenbaum, Prentice Hall Of India (2nd Edition).
2. Data Structure and Algorithm, Prof. Maria Rukadikar S.

Reference Books:

1. Data Structures, Algorithms and Applications with C++, Sahani Mc-Graw Hill.

CSE 213	DIGITAL LOGIC DESIGN *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week,	Univ. Exam: 3 Hours	**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

To introduce the basic principles for design of combinational circuit and sequential circuits. To learn simple digital circuits in preparation for computer engineering.

Course Outcomes:

A student who successfully fulfills the course requirements will have demonstrated:

1. An ability to define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.
2. An ability to understand the different Boolean algebra theorems and apply them for logic functions.
3. An ability to define the Karnaugh map for a few variables and perform an algorithmic reduction of logic functions.
4. An ability to define the following combinational circuits: multiplexer, de-multiplexers encoders/decoders, comparators, arithmetic-logic units; and to be able to build simple circuits.
5. An ability to understand asynchronous and synchronous sequential circuits, like counters and shift registers.
6. An ability to understand memories like RAM and ROM, Programmable Logic Array and Programmable Array Logic.

Syllabus:

1. **Binary Systems:** Digital Systems. Binary Numbers. Number Base Conversions. Octal and Hexadecimal Numbers. Complements. Signed Binary Numbers. Binary Codes. Binary Storage and Registers. Binary Logic
2. **Boolean Algebra and Logic Gates:** Basic Definitions. Axiomatic Definition of Boolean Algebra. Basic Theorems and Properties of Boolean Algebra. Boolean Functions. Canonical and Standard Forms. Other Logic Operations. Digital Logic Gates. Integrated Circuits.
3. **Combinational Logic Design, Gate-Level Minimization:** The Map Method. Four-Variable Map. Five-Variable Map. Product of Sums Simplification. Don't-Care Conditions. NAND and NOR Implementation. Other Two- Level Implementations. Exclusive-OR Function. Hardware Description Language (HDL).
4. **Combinational Logic:** Combinational Circuits. Analysis Procedure. Design Procedure. Binary Adder-Subtractor.DecimalAdder.BinaryMultiplier.MagnitudeComparator.Decoders. Encoders. Multiplexers. HDL For Combinational Circuits.
5. **Sequential Logic Design, Synchronous Sequential Logic**: Sequential Circuits .Latches .Flip-Flops. Analysis of Clocked Sequential Circuits. HDL For Sequential Circuits. State Reduction and Assignment. Design Procedure.
6. **Registers ad Counters:** Registers. Shift Registers. Ripple Counters. Synchronous Counters. Other Counters. HDL for Registers and Counters.
7. **Memory and Programmable Logic**: Introduction. Random-Access Memory. Memory Decoding, Error Detection and Correction. Read-Only Memory. Programmable Logic Array. Programmable Array Logic. Sequential Programmable Devices.

Text Books:

1. Digital Design, 3rdEdition, M.Morris Mano, Pearson Education.
2. Digital Logic Design, Lokesh Chaudhary& Sunil S. Chaudhary Hardeep Singh

Reference Books:

1. Digital Logic Design Principles, Norman Balabanian & Bradley Carlson, John Wiley & Sons(Asia) Pvt.Ltd.,2002
2. Fundamentals of Digital Logic with VHDL Design, Stephen Brown and ZvonkoVranesic, Tata McGraw-Hill Edition,2002

CSE214	OBJECT ORIENTED PROGRAMMING *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods+1Tut/week,	Univ. Exam: 3 Hours	**Credits: 4**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

On completing this course student will be able to

1. Understand the syntax and principles of Object oriented programming language, and to programs using control statements, classes and interfaces.
2. Design and development of secure and extendable C++ applications.
3. Understanding the concepts of oops, different predefined classes and packages
4. Understand the concepts of polymorphism

Course Outcomes:

1. Students will be able to handle I/O streams and Run time errors.
2. Students will be able to construct applications and Identify where data structures are appearing in them.

Syllabus:

1. **Basic Concepts of OOP:** Procedural Paradigms, Object Oriented Paradigm, OOP Principles and Terminology, OOP benefits, Procedure and Object Oriented programming languages, advantages and disadvantages. Introduction to U.M.L: Description of various U.M.L. Diagrams with examples.
2. **Introduction to C++ :** Basic Structure C++ Program, variable and Constants, Symbolic Constants, basic data types and derived data type, variable declaration, dynamic initialization, type modifiers, type casting, i/o statements in C++, operators and example programs**,** Control Structures- Programs using all control structures and statements, Functions: Function Prototypes, Function Components, Returning values from functions, actual and formal arguments, parameter passing methods, Inline functions.
3. **Classes and Objects:** Introduction to class, class definition, class specification, Member functions, data members, access specifiers, scope resolution operator, Object definition and creation, array of objects, pointers, Pointers to objects, this pointer, dynamic allocation operator, friend functions, const and volatile functions, static members, nested classes, local classes,
4. **Constructors and destructors:** Definition of constructor and destructor, default constructor, parameterized constructor, copy constructor, constructor with dynamic allocation, explicit constructor.
5. **Inheritance:** Definition, base class, derived class, using access specifiers in inheritance, Types of Inheritance, protected data with private inheritance, constructor in derived and base class, abstract classes.
6. **Virtual functions and Polymorphism:** Function overloading, arrays and strings, Operator overloading through unary and binary operator, Friend functions, Assignment operator, Stream operator overloading and type conversion; Virtual functions, Pure Virtual function, Dynamic polymorphism, Virtual destructor, Virtual base class, Dynamic casting, Cross casting, Down casting, Program development.
7. **Streams and Files in C++:** Stream Classes, Formatted and unformatted data, manipulators, user defined manipulators, file streams, file pointer manipulation; file open and close, file handling, random access, object serialization, name spaces, std namespaces, ANSI string objects and standard template library.
8. **Templates, Exception handling:** Class templates, Function templates, Member function templates, Exception handling - try-catch-throw paradigm, exception specification, terminate and un expected functions- uncaught exception, exception handling mechanism, multiple catch, nested try, Rethrowing the exceptions.

Text Books:

1. Object Oriented Programming through C++ by Robat Laphore.
2. Object oriented Programming using C++: E. Balagurusamy, PHI.

Reference Books:

1. Object Oriented Programming in C++: N. Barkakati, PHI
2. The Complete reference in C++ by Herbert Shieldt, TMH
3. The C++ Programming Language by B. Stroustrup, Pearson Education

IT 215	SYSTEMS PROGRAMMING	
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction to Systems Programming**: Introduction to Assembly Language Programming - Introduction to Instruction Formats, Data formats - Role of Base Register, Index Register.
2. **Introduction to Assembler**: databases used in assembler design, Design of Assembler - Single Pass & Double Pass.
3. **Introduction to Macros**: various types of Macros, Design of Macro Processor - Single Pass & Double Pass.
4. **Introduction to Loaders:** functions of a loader, types of Loaders, databases used in Loaders, Design of Loaders - Absolute & DLL.
5. **Introduction to Software Tools**: Text editors, Interpreters, Program Generators, Debug Monitors.

Text Books:

1. Systems Programming by Donovan Tata McGraw Hill

Reference Books:

1. System Programming by Dhamdhere Tata McGraw Hill, IInd Revised Edition

CSE 216	PRINCIPLES OF ECONOMICS AND MANAGEMEENT *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. Apply economic reasoning to the analysis of selected contemporary economic problems.
2. Understand how households (demand) and businesses (supply) interact in various market structures to determine price and quantity of goods and services produced and consumed.
3. Analyze the efficiency and equity implications of government interference in markets.
4. Recognize and identify situations leading to market failures and government failures.
5. Evaluate the intent and outcomes of government stabilization policies designed to correct macroeconomic problems.
6. Use economic problem solving skills to discuss the opportunities and challenges of the increasing globalization of the world economy.

Course Outcomes:

1. Understand the links between production costs and the economic models of supply.
2. Represent supply, in graphical form, including the upward slope of the supply curve and what shifts the supply curve.
3. Understand the efficiency and equity implications of market interference, including government policy.
4. Understand how different degrees of competition in a market affect pricing and output.
5. Apply economic reasoning to individual and firm behavior.

Syllabus:

1. **Introduction to Managerial Economics:** Wealth, Welfare and Scarce Definitions of Economics; micro and Macro Economics; Demand- Law of Demand, Elasticity of Demand, types of Elasticity and factors of determining price elasticity of Demand: utility- Law of Diminishing Marginal Utility and its limitations.
2. **Conditions of Different Market Structures:** Perfect Competition, Monopolistic Competition, Monopoly, Oligopoly, and Duopoly.
3. **Forms of Business Organizations:** Sole Proprietorship, Partnership, Joint Stock Company- Private Limited and Public Limited Companies, Public Enterprises and their types.
4. **Introduction to Management:** Functions of Management- Taylor's Scientific management; Henry Fayol's Principle of Management; Human Resource Management- basic Functions of HR Manager; Man Power Planning, Recruitment, Selection, Training, Development, Placement, Compensation and performance Appraisal(in brief).
5. **Production Management:** Production Planning and Control, plant Location, Break- Even Analysis, assumptions and applications.
6. **Financial Management:** Types of Capital: Fixed and Working Capital and Methods of Raining Finance; Depreciation: Straight Line and Diminishing Balance Methods. Marketing Management: Functions of marketing and Distribution Channels.
7. **Entrepreneurship:** Entrepreneurial Functions, Entrepreneurial Development: Objectives, Training, Benefits: Phases of Installing a project

Text Books:

1. K.K.DEWETT, **Modern Economic Theory**, S.Chand and Company, NewDelhi-55.
2. S.C. Sharma and Banga T. R., Industrial Organization & Engineering Economics, khanna Publications, Delhi-6.

Reference Books:

1. A.R. Arya Sri, Management Science, TMH publications, NewDelhi-20.
2. A.R. Arya Sri Managerial Economics and Financial Analysis, TMH Publications, new

CSE 217	DATA STRUCTURES LAB *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Course Objectives:

1. To implement stacks and queues using arrays and linked lists.
2. To develop programs for searching and sorting algorithms.
3. To write programs using concepts of various trees.
4. To implement programs using graphs.

Course Outcomes:

1. Student will be able to write programs to implement stacks and queues.
2. Ability to implement various searching and sorting techniques.
3. Ability to implement programs using trees and graphs.

List of Programs:

1. Write a C program for sorting a list using Bubble sort and then apply binary search.
2. Write a C program to implement the operations on stacks.
3. Write a C program to implement the operations on circular queues.
4. Write a C program for evaluating a given postfix expression using stack.
5. Write a C program for converting a given infix expression to postfix form using stack.
6. Write a C program for implementing the operations of a dequeue
7. Write a C program for the representation of polynomials using circular linked list and for the addition of two such polynomials
8. Write a C program for quick sort
9. Write a C program for Merge sort.
10. Write a C program for Heap sort
11. Write a C program to create a binary search tree and for implementing the in order, preorder, post order traversal using recursion
12. a) Write a C program for finding the transitive closure of a digraph
    b) Write a C program for finding the shortest path from a given source to any vertex in a digraph using Dijkstra's algorithm
13. a) Write a C program for finding the Depth First Search of a graph.
    b) Write a C program for finding the Breadth First Search of a graph.

CSE 218	**OBJECT ORIENTED PROGRAMMINGLAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Course Objectives:

1. To develop programs using basic OOPS concepts such as classes and objects.
2. To implement programs using Inheritance concepts.
3. To implement programs using Exception handling.
4. To develop programs using operator overloading concepts.

Course Outcomes:

1. Student will be able to use OOPs concepts.
2. Ability to apply Inheritance concepts to several problems.
3. Ability to use Exception Handling concepts.

List of Programs:

1. Write a Program in C++ that implements stack operations using classes and objects.
2. Write a Program in C++ performing complex number addition using friend functions.
3. Write a Program in C++ for complex number addition using operator overloading.
4. Write a Program in C++ to perform string operations by overloading operators.
5. Write a Program in C++ on hierarchical inheritance showing public, private and protected inheritances.
6. Write a Program in C++ for computation of student's result using hybrid inheritance.
7. Write a Program in C++ implementing bubble-sort using templates.
8. Write a Program in C++ on virtual functions.
9. Write a Program in C++ for handling PushOn Full and PopOn Empty Exceptions for a Stack.
10. Write a Program in C++ for copying one file to another file using streams.
11. Write a Program in C++ for writing and reading a class object to a file.
12. Write program in C++ to implement
    a) One catch block and all Exceptions
    b) Using Multiple Catch blocks.
13. Write a program in C++ to implement the finally block.
14. Write a program in C++ to implement pointers to a derived class and virtual base classes.
15. Write a program in C++ to implement conversion of objects between different classes using conversion functions.
16. Write a program in C++ to implement function overloading- with various data types, with different number of arguments.
17. Write a program in C++ to evaluate mixed mode expressions and implicit type conversions.
18. Write a program in C++ to show that there is ambiguity in Multiple Inheritance.
19. Write a program in C++ to implement a virtual destructor.
20. Write a program in C++ to mimic a bank management system (user logins, requests for withdraw/credit, system verifies whether enough balance is available, update the account summary, etc.)

B.TECH (IT) 2nd YEAR II-SEMESTER SCHEME OF INSTRUCTION AND EXAMINATION
With effect from 2019-2020 admitted batch

Sub Ref. No	Name of the Subject	Periods			Maximum marks			Credits
		Theory	Tutorial	Lab	Exam	Internal	Total	
CSE221	DISCRETE MATHEMATICS	3	--	--	70	30	100	3
CSE222	COMPUTER ORGANIZATION & ARCHITECTURE	3	--	--	70	30	100	3
CSE223	DATABASE MANAGEMENT SYSTEMS	3	--	--	70	30	100	3
CSE224	DESIGN ANDANALYSIS OF ALGORITHMS	3	--	--	70	30	100	3
IT225	INTERNET CONCEPTS AND JAVA PROGRAMMING	3	--	--	70	30	100	3
CSE226	ENVIRONMENTAL STUDIES	3	--	--	70	30	100	0
CSE227	DBMS LAB	--	--	3	50	50	100	1.5
IT228	INTERNET CONCEPTS AND JAVA PROGRAMMING LAB	--	--	3	50	50	100	1.5
	TOTAL	**18**	**0**	**6**	**520**	**280**	**800**	**18**

CSE 221	**DISCRETE MATHEMATICS** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. To understand mathematical arguments using logical connectives and quantifiers and verify the validity of logical flow of arguments using propositional, predicate logic and truth tables.
2. To understand about permutations and combinations.
3. To understand various types of relations and discuss various properties of the relations.
4. To study the graphs, graph isomorphism and spanning trees.
5. To study about Boolean algebra and Finite State Machines.

Course Outcomes:

At the end of the course student will be able to

1. Rewrite mathematical arguments using logical connectives and quantifiers and verify the validity of logical flow of arguments using propositional, predicate logic.
2. Identify and give examples of various types of relations and describe various properties of the relations.
3. Ability to solve problems using permutations and combinations.
4. Determine isomorphism of graphs and spanning tree of a given graph using BFS/DFS algorithms. Also determine minimal spanning tree of a given graph.

Syllabus:

1. **The Foundations-Logic and Proofs:** Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers Rules of Inference, Introduction to Proofs, Proof Methods and Strategy, Basic Structures-Sets, Functions, Sequences and Sums: Sets, Set Operations, Functions, Sequences and Summations.
2. **The Fundamentals-Algorithms, the Integers and Matrices:** Algorithms, The Growth of Functions, Complexity of Algorithms, The Integers and Division, Primes and Greatest Common Devisors, Integers and Algorithms, Applications of Number Theory, Matrices.
3. **Induction and Recursion:** Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness.
   Counting: The Basics of Counting, The Pigeonhole Principle, Permutations and Combinations, Binomial Coefficients, Generalized Permutations and Combinations, Generating Permutations and Combinations.
4. **Advanced Counting Techniques:** Recurrence Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recursion Relations, Generating Functions, Inclusion-Exclusion, and Applications of Inclusion-Exclusion.
5. **Relations:** Relations and their properties, n-ary relations, applications, Representation, closure, equivalence relations, Partial orderings.
   Graphs: Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring.
6. **Trees:** Introduction to Trees, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees.
7. **Boolean Algebra:** Boolean Functions, Representing Boolean Functions, Logic Gates, Minimization of Circuits.
8. **Modeling Computation:** Languages and Grammars, Finite-State Machines with Output, Finite-State Machines with No Output, Language Recognition, Turing Machines.

Text Books:

1. Discrete Mathematics & Its Applications with Combinatorics and Graph Theory by Kenneth H Rosen, Tata McGraw-Hill Publishing Company Ltd., New Delhi.
2. Discrete Mathematical Structures by Y.N Singh.

Reference Books:

1. Discrete Mathematics for Computer Scientists & Mathematicians by Joe L. Mott, Abraham Kandel, Theodore P. Baker, Prentice-Hall, India.
2. Discrete Mathematics by Richard Johnson Baug, Pearson Education, New Delhi.
3. Discrete and Combinatorial Mathematics by Ralph. G. Grimaldi, Pearson Education, New Delhi.

CSE 222	COMPUTERORGANIZATION AND ARCHITECTURE *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week,	**Univ. Exam: 3 Hours**	**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. To study about structure and functional components of a computer.
2. Understanding the hierarchical organization of a computer system which consists of instruction set of commands.
3. Learn about the architecture of a computer from a programming view.
4. To design a balance system that minimizes performance and utilization of all elements.

Course Outcomes:

1. Knowledge about major components of a computer such as processor, memory and I/O modules along with their interconnections internally with outside world.
2. Detailed idea about architecture of central processing unit, functions of control unit, memory, I/O devices and their issues.
3. Simple and multiple processor organization and their issues.

Syllabus:

1. **Register Transfer and Micro operations:** Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro operations, Shift Micro operations, Arithmetic Logic Shift Unit.
2. **Basic Computer Organization and Design:** Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input- Output and Interrupt, Complete Computer Description, Design of Basic Computer, Design of Accumulator Logic.
3. **Micro programmed Control**: Control Memory, Address Sequencing, Micro program Example, Design of Control Unit.
4. **Central Processing Unit:** Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer(RISC), Architecture and Programming of 8085Microprocessor.
5. **Pipeline and Vector Processing:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISK Pipeline, Vector Processing, Array Processors.
6. **Input/output Organization:** Peripheral Devices, I/O interface, Asynchronous data transfer, Modes of transfer, priority Interrupt, Direct memory access, Input-Output Processor (IOP), Serial Communication.
7. **Memory Organization**: Memory Hierarchy, Main memory, Auxiliary memory, Associate Memory, Cache Memory, and Virtual memory, Memory Management Hardware.

Text Books:

1. Computer System Architecture, M. Morris Mano, Prentice Hall of India Pvt. Ltd., Third Edition, Sept.2008.
2. Computer Architecture and Organization, P.Chakraborty.
3. Microprocessor Architecture, Programming and Applications with the 8085 by Ramesh S Gaonkar

Reference Books:

1. Computer Architecture and Organization, William Stallings, PHI Pvt. Ltd., Eastern Economy Edition, Sixth Edition, 2003.
2. Computer Organization and Architecture, Linda Null, Julia Lobur, Narosa Publications ISBN81- 7319-609-5
3. Computer System Architecture", John. P.Hayes.

CSE 223	DATABASE MANAGEMENT SYSTEMS *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. To learn the evolution of DBMS Versus File systems, data models, and layers of abstraction.
2. To understand conceptual and physical aspects of database design.
3. To learn formal and commercial query language specifications.
4. To understand concurrency control, recovery management, and other related issues.

Course Outcomes:

1. The student will understand ER-modeling for conceptual database design and relational model.
2. The student is introduced to formal and commercial query languages: Relational Algebra, calculus and SQL.
3. The student will learn schema refinement and normalization.
4. The Student understands locking protocols concurrency control, and crash recovery methods.

Syllabus:

1. **Introduction**: File system versus a DBMS, Advantages of a DBMS, Describing and Storing Data in a DBMS, The Relational model, Levels of abstraction, Data Independence, Transaction management, Structure of a DBMS.
2. **Introduction to Database Design and The Relational Model**: Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships & Relationship Sets, Additional Features of the ER Model, Conceptual Design with ER Model, Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/ Altering Tables and Views.
3. **Relational Algebra and SQL**: Preliminaries, Relational Algebra, The form of a Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers and Active Databases, Embedded SQL, Dynamic SQL, JDBC.
4. **Database Design**: Schema Refinement and Normal Forms, Introduction to Schema Refinement, Functional Dependencies, Reasoning about FD's, Normal Forms, Properties of Decomposition, Normalization, Other kinds of Dependencies.
5. **Transaction Management**: The ACID Properties, Transactions & Schedules, Concurrent Execution of Transactions, Lock-Based Concurrency Control.
6. **Concurrency Control**: 2PL, Serializability and Recoverability, Introduction to Lock Management, Lock Conversions, Dealing with Deadlocks, Specialized Locking Techniques, Concurrency Control without Locking.
7. **Crash Recovery**: Introduction to ARIES, The Log, Other Recovery-Related Structures, The Write-Ahead Log Protocol, Check pointing, recovering from a System Crash, Media Recovery.

Text Books:

1. Database Management Systems; Raghu Ramakrishnan, Johannes Gehrke 4th Edition, McGraw- Hill
2. Database Management Systems; Raghu RamaKrishnan, Johannes Gehrke.

Reference:

1. Database System Concepts; A. Silberschatz, H. Korth 5th Edition, McGraw-Hill

CSE 224	DESIGN & ANALYSIS OF ALGORITHMS *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods + 1 tutorial/week, Univ. Exam: 3 Hours		**Credits: 4**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

On completing this course student will be able to

1. Solve problems using algorithm design methods such as the greedy method, divide and conquer, dynamic programming, backtracking, and branch and bound and writing programs for these solutions
2. Analyze the asymptotic performance of algorithms.
3. Demonstrate a familiarity with major algorithms and data structures.
4. Synthesize efficient algorithms in common engineering design situations.

Course Outcomes:

1. Students will be able to justify the correctness of algorithms using inductive proofs and invariants
2. Analyze worst-case running times of algorithms using asymptotic analysis.
3. Describe various paradigms of design use them appropriately when an algorithmic design situation calls for it.
4. Students will be able to compare between different data structures. Pick an appropriate data structure for a design situation.

Syllabus:

1. **Introduction–** Fundamentals of algorithmic problem solving – important problem type. Fundamentals of analysis of algorithms and efficiency – Analysis framework– Asymptotic Notations and Basic Efficiency classes – Mathematical Analysis of Non- recursiveAlgorithms– MathematicalAnalysisofrecursiveAlgorithms–EmpiricalAnalysis of Algorithms – Algorithm Visualization.
2. **BruteForce**–SelectionSortandBubblesort–SequentialSearchandBrute–ForceString Matching – Closest Pair and Convex-Hull Problems by Brute Force – Exhaustive Search **Divide-and-Conquer**–Merge sort– Quick sort–Binary Search–Binary Tree Traversals and Related Properties – Multiplication of large integers and Strassen's Matrix Multiplication– Closest- Pair Convex-Hull Problems by Divide- and – Conquer.
3. **Decrease – and – Conquer** – Insertion Sort – Depth-First Search and Breadth-First Search-TopologicalSorting–AlgorithmsforGeneratingCombinatorialObjects–Decrease-by-a- Constant-Factor Algorithms – Variable-Size-Decrease Algorithms.
4. **Transform-and-Conquer** – Presorting – Gaussian Elimination – Balanced Search Trees– Heaps and Heap sort – Horner's Rule and Binary Exponentiation – Problem Reduction
   Space and Time Tradeoffs – Sorting by Counting – Input Enhancement in string Matching – Hashing – B-Trees
5. **Dynamic Programming** – Computing a Binomial Coefficient – Warshall's and Floyd's Algorithm – Optimal Binary Search Trees - The Knapsack Problem and Memory Functions.
6. **Greedy Technique** – Prim's Algorithm – Kruskal's Algorithm – Dijkstra's Algorithm– Huffman Trees **Limitations of Algorithm Power** – Lower-Bound Arguments –Decision Trees – P, NP and NP – complete problems – Challenges of Numerical Algorithms.
7. **Coping with the Limitations of Algorithms Power** – Backtracking – Branch-and-Bound– Approximation Algorithms for NP-hard Problems – Algorithms for solving Nonlinear Equations.

Text Books:

1. Introduction to Design & Analysis of Algorithms by Anany Levitin, Pearson Education, New Delhi, 2003
2. Fundamentals of Computer Algorithms, Horowitz and Sahni, Galgothia publications.

Reference Books:

1. Introduction to Algorithms by Thomas H. Corman, Charles E. Leiserson, Ronald R. Rivest& Clifford Stein, Prentice Hall of India, New Delhi, NewDelhi.

IT225	INTERNET CONCEPTS & JAVA PROGRAMMING	
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Fundamentals:** HTML, OOP Concepts, Comparing JAVA with C & C++, JAVA Programming language Syntax, Variables, Data types, statements and expressions.
2. **Control Statements**: If else, for, while, and do while loops, Switch statements.
3. **Arrays & Structures**: One Dimensional & Two Dimensional Arrays, Named Structures.
4. **Functions**: Parameter Passing, Static Modifier.
5. **Features of JAVA**: Classes and Interfaces, Threads and multithreaded programming, Exception handling, Introduction to packages, Math package, Lang package, Util package.
6. **Applet Programming**: Events, Event driven programming, Events like buttons, mouse, keyboards etc., Applets, Applets package, Fonts, colors, Graphics, images. AWT components, layout managers, writing event driven program using components.
7. **Networking**: Networking Basics: Socket overview, Client/Server, Reserved sockets. Proxy servers, Internet addressing; Java and the net, Inet address, TCP/IP client sockets, URL, URL connection, TCP/IP server sockets, Data grams.

Text Books:

1. Introduction to Java programming, a primar‖, Balaguruswamy.
2. Java Complete Reference, Herbt Schild.

Reference Book:

1. Introduction to Java programming‖, Daneal/Young PHI

CSE 226	ENVIRONMENTAL STUDIES *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 0**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

The Program seeks to provide students better understanding and planning for conservation through an interdisciplinary environmental science curriculum that is designed to enhance scientific inquiry and to strengthen scientific competence. Through these efforts, the Program aims at preparing and providing students to opportunities for careers in environmental sciences, environmental health, public health, and medical schools.

Course Outcomes:

1. Recognize major concepts in environmental sciences and demonstrate in-depth understanding of the environment.
2. Develop analytical skills, critical thinking, and demonstrate problem-solving skills using scientific techniques.
3. Demonstrate the knowledge and training for entering graduate or professional schools, or the job market.

Syllabus:

1. Introduction

a) Definition, Scope and importance
b) Measuring and defining environmental development: indicators

2. Ecosystem

Introduction, types, characteristic features, structure and functions of Ecosystems -Forest –Grass land - Desert -Aquatic (lakes, rivers and estuaries)

3. Environmental and Natural Resources management

a) Land resource-Land as a resource -Common property resource -Land degradation -Soil erosion and desertification -Effects of modern agriculture, fertilizer – pesticide problems
b) Forest resources Use and over-exploitation-Mining and dams- their effects on forest and tribal people
c) Water resources-Use and over-utilization of surface and ground water-Floods and droughts-Water logging and salinity-Dams –benefits and costs-Conflicts overwater
d) Energy resources
e) Energy needs-Renewable and non-renewable energy source-Use of alternate energy sources-Impact of energy use on environment

4. Bio-diversity and its conservation

a) Value of bio-diversity-consumptive and productive use, social, ethical, aesthetic and option values
b) Bio-geographical classification of India- India as a mega diversity habitat
c) Threats to biodiversity- Hot spots, habitat loss, poaching of wildlife, loss of species, seeds etc.
d) Conservation of bio-diversity- In-situ and Ex-situ conservation

5. Environmental Pollution Local and Global Issues Cause, effects and control measures of Air Pollution- Indoor air pollution- Water pollution-Soil pollution- Marine pollution-Noise pollution-Solid waste management, composting, vermin culture-Urban and industrial wastes, recycling and reuse

a) Nature of thermal pollution and nuclear hazards
b) Global Warming
c) Acid rain
d) Ozone depletion

6. Environmental problems in India Drinking water, Sanitation and Public health

a) Effects of activities on the quality of environment, Urbanization-Transportation-Industrialization-
b) Green revolution
c) Water scarcity and Ground Water depletion
d) Controversies on major dams- resettlement and rehabilitation of people: problems and concerns
e) Rain water harvesting, cloud seeding and watershed management

7. Economy and Environment The economy and environment interaction

a) Economics of development, preservation and conservation
b) Sustainability: theory and practice

c) Limits to Growth
d) Equitable use of resources for sustainable lifestyles
e) Environmental Impact Assessment

8. Social Issues and the Environment Population growth and environment

a) Environmental education
b) Environmental movements
c) Environment vs Development

9. Institutions and Governance (5lectures)

a) Regulation by Government
b) Monitoring and Enforcement of Environmental regulation
c) Environmental Acts Water (Prevention and Control of pollution) act-Air (Prevention and Control of pollution) act-Envt. Protection act-Wild life Protection act-Forest Conservation act-Coastal Zone Regulations
d) Institutions and policies relating to India
e) Environmental Governance

10. International Conventions (2lectures)

a) Stockholm Conference1972
b) Earth Summit1992
c) World Commission for environmental Development(WCED)

11. Case Studies: Chipko movement

a) Narmada Bachao Andolan
b) Silent Valley Project
c) Madhura Refinery and Taj Mahal
d) Industrialization of Pattancheru
e) Nuclear reactor in Nagarjunasagar
f) Tehri dam
g) Ralegaon Siddhi (AnnaHazzare)
h) Kollerulake-aquaculture
i) Florosis in AndhraPradesh

12. Field Work

a) Visit to a local area to document and mapping environmental assests- river/ forest/ grassland/Hill/ Mountain.
b) Study of local environment- common plants, insects, birds
c) Study of simple ecosystems- pond, river, hill, slopes etc.
d) Visit to Industries, Water treatment plants, affluent treatment plants

CSE 227	**DATABASE MANAGEMENT SYSTEMS LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Course Objectives:
1. To introduce to a commercial DBMS such as ORACLE.
2. To learn and practice SQL commands for schema creation, data manipulation.
3. To learn conceptual and physical database design based on a case study.
4. To apply database design stages by studying a case study.

Course Outcomes:
1. The student is exposed to a commercial RDBMS environment such as ORACLE.
2. The student will learn SQL commands for data definition and manipulation.
3. The student understands conceptual through physical data base design.
4. The student takes up a case study and applies the design steps.

Features of a commercial RDBMS package such as ORACLE/DB2, MS Access, MYSQL & Structured Query Language (SQL) used with the RDBMS.

I. Laboratory Exercises Should Include:

a. Defining Schemas for Applications,
b. Creation of Database,
c. Writing SQL Queries,
d. Retrieve Information from Database,
e. Creating Views
f. Creating Triggers
g. Normalization up to Third Normal Form
h. Use of Host Languages,
i. Interface with Embedded SQL,
j. Use of Forms
k. Report Writing

II. Some sample applications are given below:

1. Accounting Package for Shops,
2. Database Manager for Magazine Agency or News paper Agency,
3. Ticket Booking for Performances,
4. Preparing Greeting Cards & Birthday Cards
5. Personal Accounts - Insurance, Loans, Mortgage Payments, Etc.,
6. Doctor's Diary & Billing System
7. Personal Bank Account
8. Class Marks Management
9. Hostel Accounting
10. Video Tape Library,
11. History of Cricket Scores,
12. Cable TV Transmission Program Manager,
13. Personal Library.
14. Sailors Database
15. Suppliers and Parts Database

IT 228	INTERNET CONCEPTS & JAVA PROGRAMMING LAB	
Instruction:3Periods/week,	Univ. Exam: 3Hours	**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Lab experiments:

1. (a) Program to display the area of a rectangle.
   (b) Program to find Sum of series1+x+x2+x3+……
2. (a) Write a class to display the area of rectangle and inherit this class into other class which is displaying perimeter of a rectangle and implement.
   (b) Write a class to add three no's inherit this class into other class to add five no's and implement it.
3. (a) Write a program to print the path, filename and extension for a given path of a file.
   (b) Write a program to receive two command line arguments check whether they are equal or not.
4. (a) A program to take two arguments and divide the first argument with second argument and display the result. Display the error message if divide by zero without abnormal exit.
   (b) A program to accept more than one string and arrange them in alphabetical order.
   (c) Write a program to display simultaneously output of even and odd numbers starting from one to specified number.
5. Write a program to accept data from keyboard and write it into a file.
6. Write a java program to implement stack & Queue operations.
7. Write a program to draw line and circle using mouse.
8. Write a applet program for drawing the bar chart.
9. Write a applet program to design a calculator for implementing basic functions like +, - , *, /.
10. Write a program to check active ports in system.

B.TECH (IT) 3rd YEAR I-SEMESTER SCHEME OF INSTRUCTION AND EXAMINATION
With effect from 2019-2020 admitted batch

Sub Ref. No	Name of the Subject	Periods			Maximum marks			Credits
		Theory	Tutorial	Lab	Exam	Internal	Total	
CSE311	COMPUTER NETWORKS	3	--	--	70	30	100	3
CSE312	OPERATING SYSTEMS	3	--	--	70	30	100	4
CSE313	FORMAL LANGUAGES & AUTOMATA THEORY	3	--	--	70	30	100	3
CSE314	OBJECT ORIENTED SOFTWARE ENGINEERING	3	--	--	70	30	100	4
CSE315	OPERATIONS RESEARCH	3	--	--	70	30	100	3
IT316	ELECTIVE-I	3	--	--	70	30	100	3
IT317	SOFTWARE ENGINEERING MINI PROJECTS LAB	--	--	3	50	50	100	1.5
CSE318	OPERATING SYSTEMS LAB	--	--	3	50	50	100	1.5
CSE319	SOFT SKILLS LAB	--	--	3	50	50	100	1.5
	TOTAL	18	0	9	570	330	900	**24.5**

ELECTIVE-I:
1. File Structures
2. Principles of Programming Languages
3. Distributed Systems

CSE 311	**COMPUTER NETWORKS** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. To make the students understanding of basic requirements of network hardware, software and its architecture.
2. Familiarize the students with layered architecture of the network software and hierarchal nature of the network physical infrastructure.
3. Study of various network interconnecting devices and other associated network hardware.
4. Introduction of advanced networking concepts and wireless and wireless sensor networks.

Course Outcomes:

1. The student must be able to understand the design and estimate the requirements for practical setup of a given network scenario and size.
2. Realize the Operation, maintenance and management of the Internet by mapping the theoretical networking concepts to the real-time network scenarios.
3. Demonstrate the applications of wireless Networks and over view of advanced networking concepts.
4. Identify different networking devices and their usage and functionality.

Syllabus:

1. **Introduction to Computer Networks:** Introduction, Network Hardware, Network Software, Reference Models, Network Examples, Internet Based Applications.
2. **The Medium Access Control**: The Channel Allocation Problem, CSMA Protocols, Collision Free Protocols, The Ethernet, Wireless LANS, Bluetooth
3. **Network Layer:** Network Layer Design Issues, Routing Algorithms, Congestion Control Algorithms, Net work Layer in the Internet, IP Protocol, IP Address, Subnets, and Internetworking.
4. **Transport layer:** Transport Service, Elements of Transport Protocols, TCP and UDP Protocols, Quality of Service Model, Best Effort Model, Network Performance Issues.
5. **Application Layer:** Over View of DNS, SNMP, Electronic Mail, FTP, TFTP, BOOTP, HTTP Protocols, World Wide Web, Firewalls.
6. **Network Devices:** Over View of Repeaters, Bridges, Routers, Gateways, Multiprotocol Routers, Hubs, Switches, Modems, Channel Service Unit CSU, Data Service Units DSU, NIC, Wireless Access Points, Transceivers, Firewalls, Proxies.
7. **Overview of** Cellular Networks, Ad-hoc Networks, Mobile Ad-hoc Networks, Sensor Networks

Text Books:

1. Computer Networks, Andrews S Tanenbaum,, 5th Edition, Pearson Edu.
2. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education.

References Books:

1. Data Communications and Networking ,Behrouz A Forouzan , Tata McGraw-Hill Co Ltd, Second Edition, ISBN: 0-07-049935-7
2. Computer networks, Mayank Dave, CENGAGE.
3. Computer networks, A system Approach, 5thed, Larry L Peterson and Bruce S Davie, Elsevier.
4. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson.

CSE312	**OPERATING SYSTEMS** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 4**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course objectives:

1. To understand evolution of Operating System.
2. To understand operating system as a layer of abstraction above physical hardware that facilitates usage convenience and efficient resource management of computer system resources.
3. To learn design and implementation of policies and mechanisms for OS subsystem.
4. To investigate case studies to understand the design philosophies / paradigm for popular multiuser or single user operating system.

Course Outcomes:

1. The student understands OS evolution, its structure and services provided by it.
2. Learn process life cycle, process scheduling objectives, policies and mechanisms, process synchronization; inter process communication, deadlocks and other process subsystem related concepts.
3. Learn memory hierarchy, allocation and deallocation policies and mechanism for main and auxiliary memory; file system design and implementation issues.
4. Investigate UNIX/ LINUX and Windows OS platforms w.r.t similarities and differences in design philosophies.

Syllabus:

1. **Introduction to Operating Systems**: Over View of Operating Systems, Types of Operating Systems, Operating System Structures, Operating System Services, System Calls, Virtual Machines, Operating System Design and Implementation.
2. **Process Management**: Process Concepts, Operations on Processes, Cooperating Processes, Threads, Inter Process Communication, Process Scheduling, Scheduling Algorithms, Multiple- Processor Scheduling, Thread Scheduling.
3. **Process Synchronization:** The Critical Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors.
4. **Deadlocks**: System Model, Deadlock Characterization, Methods For Handling Deadlocks, Deadlock Prevention, Avoidance, Deadlock Detection, Recovery from Deadlocks
5. **Memory Management:** Logical versus Physical Address, Swapping, contiguous memory allocation, paging, structure of the page table , segmentation, Virtual Memory, Demand Paging, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped files
6. **FileSystems,Implementation,andSecondary-storageStructure:**Conceptofafile,Access Methods, Directory Structure, Protection, File System Structure, Allocation Methods, Free Space Management, Directory Management, Device Drivers, overview of Mass-storage structure, Disk structure, disk attachment, disk scheduling, swap-space management.
7. **Case study:** Overview of LINUX, Windows Operating systems

Text Books:

1. Operating Systems, Abraham Silbers chatz, Peter Baer Galvin, and Greg Gagne, John Wiley Publ., Seventh Edition.
2. Operating Systems; A Practical Approach. Rajiv Chopra**.**

Reference Books:

1. Modern Operating Systems, Andrew S. Tanenbaum,2nd edition, 1995,PHI.
2. Operating Systems, William Stallings 5th Edition -PHI
3. Operating Systems: A Design-Oriented Approach', Charles Crowley, 'Tata Hill Co., 1998 edition.

CSE 313	**FORMAL LANGUAGES & AUTOMATA THEORY** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits:3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course objectives:

1. To introduce the concepts in automata theory and theory of computation to design grammars and recognizers for different formal languages.
2. To employ finite state machines to solve problems in computing.
3. To introduce finite state machines, context free grammars and Turing Machines and their properties as the basis for the formal expressivity of computer languages for solving linguistic decision problems.
4. To understand the concepts of tractability and decidability, the concepts of NP-completeness and NP-hard problem and also the challenges for Theoretical Computer Science and its contribution to other sciences.

Course outcomes:

1. Ability to think analytically and intuitively for problem-solving situations in related areas of theory in computer science
2. Ability to describe the language accepted by an automata or generated by a regular expression or a context-free grammar;
3. Ability to Understand the functioning of Finite-State Machines, Deterministic Finite-State Automata, Nondeterministic Finite-State Automata and Pushdown Automata and Turing Machines.

Syllabus:

1. **Definitions of alphabet**, strings, language, grammar, types of grammar, types of machines, generation of languages from grammar, construction of grammar from the given description of languages.
2. **Definition of finite state machine**, Definite state machine, indefinite state machine, representations in mathematical diagram, tabular etc., id of finite state machine's, design of finite state machine from the given description, elimination of e-transitions , indefinite state machine to definite state machine, optimization of finite state machine.
3. **Conversion of regular grammar to finite state machine**, finite state machine to regular grammar, discussion of pumping lemma, systematic way of construction of finite state machine.
4. **Definition of regular expression**, regular algebra, minimization of regular expressions, closure properties, construction of regular expression from the given description, regular expression to finite state machine, finite state machine to regular expression, construction of regular expression for the given finite state machine- a systematic way using Arden's theorem
5. **Definition of push down machine**, push down machine, types of push down machine's, push down machine to context free grammar, context free grammar to push down machine, design methodology of various push down machine's, push down machine by empty stack, push down machine by final states, conversion from one type to other type, applications of push down machine's.
6. **Parsing tree**, bottom-up parsing, top-down parsing, types of context free grammar's, left-most and right most derivations, productions, reductions, optimization of context free grammar's, elimination of e productions, unit productions, normal forms- cnf, gnf.
7. **Definition of Turing machine**, ways of representing Turing machine's- tabular form, diagram, mathematical form, quintuples etc., design of Turing machine, id of Turing machine, types of Turing machine, halting problem, church's thesis, universal Turing machine, Gödel number, definitions of recursive functions- prf, rf, decidability.

NOTE: Theorem proofs are eliminated

Text Books:

1. Introduction to automata theory, languages and computation, John.E.H.P croft/ Rajeev Motwani& JD Ullman—pearson education- III edition
2. Theory of computation, K.L.P.Mishra and N.Chandrasekhar, PHI

Reference Books:

1. Theory of computation, formal languages and automata theory, G P SaradhiVarma, B.ThirupathiRao – Sci Tech publications.

CSE 314	OBJECT ORIENTED SOFTWARE ENGINEERING *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods+1Tut/week, Univ. Exam: 3 Hours		**Credits: 4**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course objectives:

1. To explain the importance of OOSE in Software development.
2. To explain the students the importance of Requirements Engineering.
3. To explain the role of UML and Testing in Software Development.
4. To explain the entire Software Development Process with aid of case studies.

Course Outcomes:

1. Ability to define a problem and perform Requirements Engineering.
2. Ability to draw UML diagrams for the requirements gathered.
3. Ability to implement the designed problem in Object Oriented Programming Language and
4. Test whether all the requirements specified have been achieved or not.

Syllabus:

1. **Introduction to Object Oriented Software Engineering:** Nature of the Software, Types of Software, Software Engineering Projects, Software Engineering Activities, Software Quality, Introduction to Object Orientation, Software Process Models-Waterfall Model, Opportunistic Model, Phased Released Model, Spiral Model, Evolutionary Model, Concurrent Engineering Model.
2. **Requirements Engineering:** Domain Analysis, Problem Definition and Scope, Requirements Definition, Types of Requirements, Techniques for Gathering and Analyzing Requirements, Requirement Documents, Reviewing, Managing Change in Requirements.
3. **Unified Modeling Language & Use Case Modeling:** Introduction to UML, Modeling Concepts, Types of UML Diagrams with Examples; User-Centered Design, Characteristics of Users, Developing Use-Case Models of Systems, Use-Case Diagram, Use- Case Descriptions, Basics of User Interface Design, Usability Principles, User Interfaces.
4. **Class Design and Class Diagrams:** Essentials of UML Class Diagrams, Associations and Multiplicity, Other Relationships, Generalization, Instance Diagrams, Advanced Features of Class Diagrams, Interaction and Behavioral Diagrams: Interaction Diagrams, State Diagrams, Activity Diagrams, Component and Deployment Diagrams.
5. **Software Design and Architecture:** Process of Design, Principles Leading to Good Design, Techniques for Making Good Design Decisions, Good Design Document; Pattern Introduction, Design Patterns: Abstraction-Occurrence Pattern, General Hierarchical Pattern, Play-Role Pattern, Singleton Pattern, Observer Pattern, Delegation Pattern, Adaptor Pattern, Façade Pattern, Immutable Pattern, Read-Only Interface Pattern and The Proxy Pattern; Software Architecture Contents of Architecture Model, Architectural Patterns: Multilayer, Client-Server, Broker, Transaction Processing, Pipe& Filter and MVC Architectural Patterns.
6. **Software Testing:** Overview of Testing, Testing Concepts, Testing Activities, Testing Strategies, Unit Testing, Integration Testing, Function Testing, Structural Testing, Class Based Testing Strategies, Use Case/Scenario Based Testing, Regression Testing, Performance Testing, System Testing, Acceptance Testing, Installation Testing, OO Test Design Issues, Test Case Design, Quality Assurance, Root Cause Analysis, Post-Mortem Analysis.
7. **Software Process Management:** Introduction to Software Project Management, Rationale Management, Configuration Management, Activities of Software Project Management, Structure of Project Plan, Software Engineering Teams, Software Cost Estimation, Project Scheduling, Tracking and Monitoring.

CASESTUDY:

1. Simple Chat Instant Messaging System
2. GPS Based Automobile Navigation System
3. Waste Management Inspection Tracking System(WMITS)
4. Geographical Information System

Text Books:

1. Object-Oriented Software Engineering Practical software development using UML and Java by Timothy C. Lethbridge & Robert, Langaniere Mcgraw-Hill
2. Software Engineeing, K.K. Agarwal, New Age Publications2008
3. Object-Oriented Software Engineering: Using UML, Patterns and Java, Bernd Bruegge and Allen H. Dutoit, 2nd Edition, Pearson Education Asia.

Reference Books:

1. Software Engineering: A Practitioner's Approach, Roger S Pressman.
2. A Practical Guide to Testing Object-Oriented Software, John D. McGregor; David A.Sykes, Addison-Wesley Professional.

CSE 315	OPERATIONS RESEARCH *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives

1. To discuss about basic Operation Research concepts, Formulation of LPP and its solution using graphical method.
2. To discuss about standard form of LPP. Solving LPP using various methods.
3. To study the various solutions of transportation problems and assignment problems.
4. To discuss about PERT and CPM charts
5. To discuss about replacement problems, inventory problems and game theory.

Course Outcomes:

1. Ability to solve LPP problems using various methods.
2. Ability to solve transportation and assignment problems using several methods.
3. Analyze the PERT and CPM charts.
4. Ability to solve replacement problems and game theory problems.

Syllabus:

1. **Overview of Operations Research**, Types of OR Models, Phases of Operations Research–OR Techniques, Introduction to Linear Programming, Formulation of Linear Programming Problem, Graphical Solution; Graphical Sensitivity Analysis.
2. **Standard Form of LPP**, Basic Feasible Solutions, Unrestricted Variables, Simplex Algorithm, Artificial Variables, Big M Method , Two Phase Simplex Method, Degeneracy, Alternative Optimal, Unbounded Solutions, Infeasible Solutions, Primal And Dual Problems And Their Relations, Dual Simplex Method.
3. **Transportation Problem as LPP**, Initial Solutions, North West Corner Rule, Lowest Cost Method, Vogel's Approximation Method, Optimum Solutions of TPP, Degeneracy in Transportation, Transportation Algorithms ,
4. **Assignment Problem** , Assignment Problem as LPP, Hungarian Method, Travelling Salesman Problem, Solutions Of TSP, Sequencing Problems, N-Jobs Two Machine Problems, N-Jobs K Machines Problems, Two-Jobs M-Machine Problems, Crew Scheduling Problems
5. **Network Representation of A Project** ,CP Mand PERT, Critical Path Calculations, Time– Cost Optimizations, PERT Analysis and Probability Considerations, Resource Analysis in Network Scheduling.
6. **Replacement Problems**- Individual And Group Replacement Policy, Reliability &System Failure Problems, Inventory-Factors Effecting Inventory-EOQ, Inventory Problems With and Without Shortages, Inventory Problems With Price Breakups, Multi Item Deterministic Problems. Probabilistic Inventory Problems.
7. **Game Theory**: Two Person Zero Sum Games, Mixed Strategy Games and Their Algorithms.

Text Books:

1. Operations Research, Kanti Swaroop, P.K. Gupta, Man Mohan, Sulthan Chand& Sons Education
2. Publishers Operations Research– An Introduction, Handy A Taha–Pearson Education.

Reference Books:

1. Operations Research Panneer Selvan Prentice Hall Of India.
2. Operations Research By S.DSharma
3. Introduction To Operations Research, F.S. Hiller, G.J. Liberman,TMH
4. Operations Research, Richard Bronson, Schaum's Series, Mcgrawhill

IT 316	ELECTIVE-I	FILE STRUCTURES
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **File Processing Operations:** Physical and logical files, opening, reading & writing and closing files in C, seeking and special characters in files, physical devices and logical files, file-related header files in C.
2. **Secondary Storage:** Disks – organization, tracks, sectors, blocks, capacity, non-data overhead, cost of a disk access, Magnetic Tape – types, performance, organization estimation of tape length and data transmission times, disk vs tape, CD-ROM – CD- ROM as a file structure, physical organization, strengths and weakness of cd- roms, storage hierarchy
3. **Byte Journey and buffer Management:** File manager, I/O buffer, I/O processing, buffer strategies and bottlenecks
4. **File Structure Concepts:** A stream file, field structures, reading a stream of fields, record structures and that uses a length indicator, Mixing numbers and characters – use of a hex dump, reading the variable length records from the files.
5. **Managing records in C files:** Retrieving records by keys, sequential search, direct access, choosing a record structure and record length, header records, file access and file organization
6. **Organizing files for performance:** Data compression, reclaiming space – record deletion and storage compaction, deleting fixed-length records for reclaiming space dynamically, deleting variable-length records, space fragmentation, replacement strategies.
7. **Indexing,** Index, A simple index with an entry sequenced file, basic operations on an indexed, entry sequenced file, indexes that are too large to hold in memory, indexing to provide access by multiple keys, retrieval using combination of secondary keys, improving the secondary index structure – inverted lists
8. **Indexed sequential file access and prefix B^+Trees:** Indexed sequential access, maintaining a sequence set, adding a simple index to the sequence set, the tree, simple prefix B content of the index: separators instead of keys, the simple prefix B tree maintenance, index set block size, internal set block size, internal structure of index set blocks: a variable B^+ tree order B- tree, loading a simple prefix
9. **Special Note: Implementation in C only Hashing:** Collisions in hashing, a simple hashing algorithms, hashing functions and record distributions, memory requirements, collision resolution by progressive overflow, buckets, deletions
10. **Extendable hashing:** Working of extendable hashing, implementation, deletion, extendable hashing performance
11. **Designing file structure for CD-ROM:** Tree structure on CD-ROM, hashing files on CD-ROM, CD-ROM file structure

Text Books:

1. File Structures – An Object Oriented Approach with C++ by Michael J. Folk, BillZoellick and Greg Riccardi,, Pearson

IT 316	ELECTIVE-I **PRINCIPLES OF FPROGRAMMING LANGUAGES**	
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course objectives:

1. To learn the underlying principles and concepts of programming language.
2. To understand programming language translation process.
3. To expose students to the important paradigms of programming.
4. To understand the concepts of distributed processing and network programming.

Course outcomes:

1. Ability to compare different programming languages.
2. Ability to discuss the significant achievements in programming language history.
3. Ability to assess the programming languages in scientific manner.

Syllabus:

1. **Language design issues:** Study programming Languages, History of programming Languages, role of programming Languages, Programming Environments.
2. **Impact of Machine Architectures**: Operation of a Computer, Virtual Computers and Binding Times**;** Language Translation Issues: Programming Language Syntax, Stages in Translation, Formal Translation Models, Recursive Descent Parsing; Modeling Language Properties: Formal Properties of Languages, Language Semantics.
3. **Elementary Data Types**: Properties of Types and Objects, Scalar Data Types, Composite Data Types **Encapsulation**: Structured Data Types, Abstract Data Types, Encapsulation by Subprograms, Type Definitions .**Inheritance**: Abstract Data Types Revisited ,Inheritance, Polymorphism
4. **Sequence Control**: Implement and Explicit Sequence Control, Sequence with Arithmetic Expressions, Sequence Control Between Statements, Sequencing with Non-arithmetic Expressions.
5. **Subprogram Control**: Subprogram Sequence Control Attributes of Data Control, Parameter Transmission, Explicit Common Environment.
6. **Storage Management**: Elements Requiring Storage, Programmer- and System - Controlled Storage, Static Storage Management, Heap Storage Management
7. **Distributed Processing**: Variations on Subprogram Control, Parallel Programming, Hardware Developments, and Software Architecture. **Network Programming**: Desktop Publishing, The World Wide Web.

Text Books:

1. Programming languages – Design and Implementation by Terrence W. Pratt Marvin V. Zelkowitz.3 rd Edition, Prentice Hall of India.

Reference Books:

1. Concepts of Programming Languages by Robert L. Sebesta, 4thEdition ,PearsonEducation.
2. Fundamentals of Programming Languages, Design & Implementation by SeyedH .Roosta. Vikas publications.
3. Programming Languages by Paradigm and Practice – Doris Appleby Julius J. Vendekopple Tata McGraw Hill Edition.

IT 316	ELECTIVE-I	DISTRIBUTED SYSTEMS
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

This course provides an introduction to the fundamentals of distributed computer systems, assuming the availability of facilities for data transmission.

Course Outcomes:

By the end of the course, students should be able to build distributed systems that:

1. Scale as the number of entities in the system increase
2. Can sustain failures and recover from them
3. Work with distributed, fault tolerant file systems
4. Can handle and process large data volumes
5. Are secure and handle certain classes of distributed denial of service attacks
6. Are Loosely coupled, transactional and eventually stable

Syllabus:

1. **Introduction to Distributed Systems**, What is a Distributed System?, Hard ware concepts, Software concepts, Design issues.
2. **Communication in Distributed Systems,** Lay red Protocols, ATM networks, The Client – server model, Remote Procedure call, and Group communication.
3. **Synchronization in Distributed System**, Clock Synchronization, Mutual Exclusion, Election algorithms, Atomic transactions, Deadlocks in Distributed Systems.
4. **Process and processors in Distributed System** threads, System Models, Processors allocation, Scheduling in Distributed System, Fault tolerance, Real time Distributed System.
5. **Distributed File Systems**, Distributed File System Design, Distributed File System implementation, Trends in Distributed File System.
6. **Distributed Shared Memory**, Introduction, What is Shared memory?, Consistency models, Page based Distributed Shared memory, Shared – variable Distributed Shared memory, Object based Distributed Shared Memory.

Text Book:

1. Distributed Operating Systems, Andrew S. Tanenbanm

Reference Book:

1. Advanced Concepts in Operating Systems, Makes Singhal and NiranjanG.Shivaratri

IT317	SOFTWARE ENGINEERING MINI PROJECT LAB	
Instruction: 3Periods/week,	Univ. Exam: 3Hours	**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

The purpose of the Software Engineering Lab course is to familiarize the students with modern software engineering methods and tools, Rational Products. The course is realized as a project-like assignment that can, in principle, by a team of three/four students working full time. Typically the assignments have been completed during the semester by each project team.

The goal of the Software Engineering Project is to have a walk through from the requirements, design to implementing and testing. An emphasis is put on proper documentation. Term projects are projects that a group student might take through from initial specification to implementation by giving equal importance to both design and implementation.

Cycle I: Practicing UML diagrams using IBM Rational Rose. 6*3 periods= 18periods

Before developing a mini-project, in this cycle, the student is acquainted with different UML diagrams using Rational Rose. The experiments should include drawing UML diagrams listed below for two demo/example applications assigned by the lab Instructor. The input for the following experiments is problem statement for any two demo projects supplied by the instructor.

1. Introduction to Rational Rose and Practicing the following diagrams
   a. Activity diagrams for the overall business process of the projects
   b. Use-case diagram for the demo projects along with Use-case descriptions and sub-diagrams for Use-cases.
2. Class diagram- Class diagrams including the features like classes, relationships, attributes and methods along with their visibilities.
3. Interaction diagrams- Sequence diagrams and Collaboration diagrams for different scenarios of the systems with all features like actors, objects and interactions.
4. Activity diagrams, State chart and other diagrams - Activity diagrams including the features like fork join and swim lanes. State diagrams including composite states and transitions. Component diagrams, Package diagrams and Deployment diagrams.
5. Forward and Reverser Engineering- Forward Engineering Class diagrams to classes in C++ and java and persistent classes to a database. Reverse Engineering C++ code, java code and a database.
6. Documentation using Rational Rose clear quest.

Cycle II: Mini-Project 8*3 periods=24periods

The project deliverables include

- Problem statement
- Requirements Analysis
- Design
  - A Software Design Description and a System Design.
  - A test specification.
- Implementation
  - Implement the assigned project with one of the following web technologies
    - **Front end:** Java technologies/PHP/MS.NET Technologies
    - **Backend:** Oracle/My-SQL/SQL-Server
- Testing

References:

1. Project-based software engineering: An Object- oriented approach, Evelyn Stiller, Cathie LeBlanc, Pearson Education
2. Visual Modeling with Rational Rose 2002 and UML, Terry Quatrini, Pearson Education

CSE 318	**OPERATING SYSTEMS LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Course Objectives:

1. To learn about UNIX/LINUX operating system, its intervals.
2. To learn system programming for UNIX/LINUX Operating System.
3. To understand UNIX/LINUX shell and its programming.
4. To understand resource management policies and mechanisms and their performance evaluation.

Course Outcomes:

1. The student practices UNIX commands, Vi editor, shell commands.
2. The student develops skill in writing C programs using system calls for process management, inter process communication and other aspects.
3. The student learns shell programming and develops skill for writing scripts for batch level tasks.
4. The student learns to simulate OS resource management aspects like process scheduling, page replacement and others to evaluate performance.

Module I

1. OS lab familiarization, Home Assignment on Unix commands, Vi editor
2. Simple C programs using command line arguments, system calls, library function calls, make utility
3. C programs using fork system call to create processes and study parent, child process mechanism
4. C programs to create process chaining, spawning
5. C programs to handle errors using errno, perror() function
6. C programs to use pipe system call for inter process communication

Module II

1. Familiarization of Unix shell programming
2. Simple shell programming exercises
3. Shell programming using decision making constructs
4. Shell programming using loop constructs
5. Shell programming for file and directory manipulation

Module III

1. C programs to study process scheduling implementing FCFS, Shortest Job First, and Round Robin algorithms
2. C programs to study page replacement implementing FIFO, Optimal, and LRU page replacement algorithms
3. C programs to study deadlock avoidance and detection
4. C Programs to simulate free space management

References:

1. UNIX concepts and applications by Sumitabha Das, TMH Publications.
2. UNIX programming by Stevens, Pearson Education.
3. Shell programming by Yashwanth Kanetkar.
4. Operating System Concepts by Silberschatz, and Peter Galvin.

CSE319	**SOFT SKILLS LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits:1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

1. English Language Skills
2. Spoken English Skills
3. Presentation Skills

B.TECH (IT) 3rd YEAR II-SEMESTER SCHEME OF INSTRUCTION AND EXAMINATION
With effect from 2019-2020 admitted batch

Sub Ref. No	Name of the Subject	Periods			Maximum marks			Credits
		Theory	Tutorial	Lab	Exam	Internal	Total	
CSE321	COMPILER DESIGN	3	--	--	70	30	100	3
CSE322	WEB TECHNOLOGIES	3	--	--	70	30	100	3
IT323	ELECTIVE-II	3	--	--	70	30	100	3
IT324	ELECTIVE-III	3	--	--	70	30	100	3
CSE325	DATA WAREHOUSING & DATA MINING	3	--	--	70	30	100	3
IT326	PROJECT-I	--	--	4	50	50	100	2
IT327	NETWORK PROGRAMMING LAB	--	--	3	50	50	100	1.5
CSE328	WEB TECHNOLOGIES LAB	--	--	3	50	50	100	1.5
	TOTAL	15	0	10	500	300	800	**20**

ELECTIVE-II
1. Computer Graphics & Multimedia
2. Information Retrieval
3. Soft Computing

ELECTIVE-III
1. Artificial Intelligence
2. Image Processing
3. Mobile Computing

CSE 321	COMPILER DESIGN *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course objectives:

1. To explain the basic understanding of grammars and language definition and introducing various phases of designing a compiler.
2. To make the student to understand the concepts underlying the design and implementation of language processors and its mechanisms.
3. To extend the knowledge of parser by parsing LL parser and LR parser.
4. To enrich the knowledge in various phases of compiler ant its use, code optimization techniques, loop optimization techniques, machine code generation, and use of symbol table.

Course outcomes:

1. Ability to design & conduct experiments for Intermediate Code Generation in compiler.
2. Ability to learn the new code optimization techniques to improve the performance of a program in terms of speed & space.
3. Ability to acquire the knowledge of modern compiler & its features.

Syllabus

1. **Introduction:** Introduction to Compilers and Language processors, Programming Language basics, Structure & Different Phases of a Compiler, Review of Compiler Structure, Structure of Optimizing Compilation, Compiler construction tools, Boot strapping, Cross compilers.
2. **Finite Automata & Lexical Analysis:** Introduction to Lexical Analysis, Lexical Analyzers, Approaches to design Lexical Analyzers, Language for specifying lexical analyzers, Introduction to Finite automata, Regular Expressions & Languages, Recognition of Tokens, Transition Diagrams, Implementation of lexical analyzers, Lexical Analyzer Generator LEX.
3. **Syntax Analysis:** Syntactic Specification of Programming Languages, Context Free Grammars & Languages, Introduction to Parsers. Top-down parsing techniques: Brute force parsing, Recursive Descent Parsing, Predictive Parsing, Bottom–up Parsing: Shift reduce parsing, Operator parsing, LR(k) parsing.
4. **Semantic Analysis and Intermediate Code Generation:** Semantic Actions, Syntax Directed Translations, Translation on the parse Tree, Implementation of Syntax Directed Translator, Intermediate Codes, Syntax Directed translation to Postfix code, Syntax Trees, Intermediate Code Generation, Three Address Code- Translation of Expressions, Type Checking & Type Conversions.
5. **Code Optimization:** Principal sources of Code Optimization, Loop Optimization, Basic Blocks & Flow Graphs, DAG Representation of Basic Blocks, Applications of DAG, Local Optimization, Unreachable Code Elimination, Dead Code Elimination, Data Flow Analysis, Data Flow Equations & Computations, Peep-Hole Optimization. Machine Dependent Optimizations, Overview of Informal Compiler Algorithm Notation (ICAN), If Simplification, Loop Simplification, Loop Inversion, Branch Optimization and Prediction
6. **Code Generation and Code Scheduling:** Issues in Code Generation, Input to Code Generator, Instruction Selection, Register Allocation, Simple Target Machine Model, Program and Instruction Costs, Register allocation & Assignments, Code Generation Algorithm, Code Generators, Optimal Code Generation for Expressions, Code Generation From DAG.
7. **Symbol Tables, Runtime Environment and Error Handling**: Contents of a Symbol Table, Data Structures for Symbol Tables; Run time Environments, Implementation of a simple Stack allocation, Heap Management, Block Structured Languages; Error Detection & Recovery, Lexical Phase Errors, Syntactic & Semantic Errors, Error Handling Routines.

Text Books:

1. Principles of Compiler Design by Aho,D. Ullman, Lam and Ravi Sethi, Pearson Education Second Edition
2. Advanced Compiler Design and Implementation, Steven Muchnic, Elsevier Publications.

Reference Books*:*

1. Compiler Construction by Kenneth. C. Louden, Vikas Pub. House.
2. Compiler Design, A.A. Pentambekar, Technical Publications
3. Modern Compiler Design, Grune.D, Van Reeuwijk K, Bal H.E, Jacobs C J H, Langendoen K,Springer,

CSE 322	**WEB TECHNOLOGIES** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

On completing this course student will be able to

1. Understand the principles of Web based application development.
2. Design dynamic content in Web Pages using JavaScript.
3. Understanding the concepts of java Servlets, java Server Pages and design applications using them.
4. Understand the concepts of Component development and design applications by establishing connections to Databases

Course Outcomes:

1. Students will be able to construct web based applications and Identify where data structures are appearing in them.
2. Students will be able to connect java programs to different databases.
3. Students will be able to develop EJB programs

Syllabus:

1. **Introduction to HTML**, Core Elements, Links and Addressing, Images, Text, Colors and Background, Lists, Tables and Layouts, Frames, Forms, Cascading Style Sheets.
2. **Introduction to Java Scripts**, Elements of Objects in Java Script, Dynamic HTML with Java Script.
3. **Document type definition**, XML Syntax, XML Schemas, Document Object model, Presenting XML, Using XML Processors.
4. **Introduction to Servlet**, Servlet Life Cycles, Servlet Basics, Tomcat Web Server, Configuring Apache Tomcat, Handling Client Request and Response, Handling Cookies, Session Tracking.
5. **Introduction to PHP**, Language Basics, Functions, Strings, Arrays.
6. **Web Techniques**, Data bases, Graphics, PDF, Dates and Times.
7. **MYSQL Installation**, Accessing MySQL Using PHP, Form Handling, Cookies, Sessions, and Authentication, Tables, Inserting Data into Tables , Selecting Data from a Table, Updating Table Deleting data from Table, Webpage creation.

Text Books:

1. Web Programming, building internet applications, 2nd Ed., Chris Bates, Wiley Dreamtech
2. Web Technologies; by Uttam K. Roy
3. The complete Reference HTML and DHTML, Thomas A. Powey

Reference Books:

1. Internet, World Wide Web , How to program, Dietel , Nieto, PHI/PEA
2. Web Tehnologies, Godbole, kahate, 2nd Ed., TMH

IT 323	ELECTIVE-II COMPUTER GRAPHICS &MULTIMEDIA	
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objective:

- To study the graphics techniques and algorithms.
- To study the multimedia concepts and various I/O technologies.
- To enable the students to develop their creativity

Course Outcomes:

1. Students will have an appreciation of the history and evolution of computer graphics, both hardware and software. Assessed by written homework assignment.
2. Students will have an understanding of 2D graphics and algorithms including: line drawing, polygon filling, clipping, and transformations.
3. Students will understand the concepts of and techniques used in 3D computer graphics, including viewing transformations, hierarchical modeling, color, lighting and texture mapping.
4. Students will be able to use a current graphics API (OpenGL). Assessed by programming assignments.

Syllabus:

1. **Output Primitives:** Introduction - Line - Circle and Ellipse Drawing Algorithms – Attributes – Two-Dimensional Geometric Transformations – Two-Dimensional Clipping and Viewing.
2. **Three-Dimensional Concepts:** Three-Dimensional Object Representations – Three-Dimensional Geometric and Modeling Transformations – Three-Dimensional Viewing – Color models – Animation.
3. **Multimedia Systems Design:** An Introduction – Multimedia applications – Multimedia System Architecture - Evolving technologies for Multimedia – Defining objects for Multimedia systems - Multimedia Data interface standards - Multimedia Databases.
4. **Multimedia File Handling**: Compression & Decompression – Data & File Format standards – Multimedia I/O technologies - Digital voice and audio – Video image and animation – Full motion video – Storage and retrieval Technologies.
5. **Hypermedia:** Multimedia Authoring & User Interface – Hypermedia messaging - Mobile Messaging – Hypermedia message component – Creating Hypermedia message – Integrated multimedia message standards – Integrated Document management – Distributed Multimedia Systems.

Text Books:

1. Donald Hearn and M.Pauline Baker, “Computer Graphics C Version”, Pearson Education, 2003. (UNIT I : Chapters 1 to 6; UNIT 2: Chapter 9 – 12, 15,16)
2. Prabat K Andleigh and KiranThakrar, “Multimedia Systems and Design”, PHI, 2003. (UNIT 3 to5)

Reference Books:

1. Judith Jeffcoate, “Multimedia in practice technology and Applications”, PHI,1998.
2. Foley, Vandam, Feiner, Huges, “Computer Graphics: Principles & Practice”, Pearson Education, second edition2003.

IT 323	ELECTIVE-II	INFORMATION RETRIEVAL
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction to Information storage and retrieval systems**: Domain Analysis of IR systems, IR and other types of Information Systems, IR System Evaluation, Introduction to Data structures and algorithms related to Information Retrieval: Basic Concepts, Data structures, Algorithms.
2. **Inverted Files**: Introduction, Structures used in Inverted Files, Building an Inverted files using a sorted array, Modifications to the Basic Techniques.
3. **Signature Files**: Introduction, Concepts of Signature files, Compression, Vertical Partitioning, Horizontal Partitioning.
4. **New Indices for Text**: PAT Trees and PAT Arrays: Introduction, PAT Tree structure, Algorithms on the PAT Trees, Building PAT Trees as PATRICA Trees, PAT representation as Arrays.
5. **Lexical Analysis and Stop lists**: Introduction, Lexical Analysis, Stop lists.
6. **Stemming Algorithms**: Introduction, Types of Stemming algorithms, Experimental Evaluations of Stemming, Stemming to Compress Inverted Files.
7. **Thesaurus Construction**: Introduction, Features of Thesauri, Thesaurus Construction, Thesaurus construction from Texts, Merging existing Thesauri.
8. **String Searching Algorithms**: Introduction, Preliminaries, The Naive Algorithm, The Knutt-Morris-Pratt Algorithm, The Boyer-Moore Algorithm, The Shift-Or Algorithm, The Karp-Rabin Algorithm.

Reference Books:

1. Modern Information Retrieval, Ricardo Baeza-Yates, Neto, PEA,2007.
2. Information Storage and Retrieval Systems: Theory and Implementation,Kowalski, Gerald, Mark Academic Press,2000.
3. Information Retrieval: Algorithms and Heuristics , Grossman, OphirFrieder,2/e, Springer,2004.
4. Information Retrieval Data Structures and Algorithms , Frakes, Ricardo Baeza-Yates, PEA
5. Information Storage and Retrieval, Robert Korfhage, John Wiley &Sons.
6. Introduction to Information Retrieval, Manning, Raghavan, Cambridge University Press.

IT 323	ELECTIVE II	SOFT COMPUTING
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Soft Computing**: Introduction to Fuzzy Computing, Neural Computing, Genetic Algorithms, Associative Memory, Adaptive Resonance Theory, Different Tools and Techniques, Usefulness and Applications.
2. **Fuzzy Sets and Fuzzy Logic:** Introduction, Fuzzy Sets Versus Crisp Sets, Operations on Fuzzy Sets, Extension Principle, Fuzzy Relations and Relation Equations, Fuzzy Numbers, Linguistic Variables, Fuzzy Logic, Linguistic Hedges, Applications,
3. **Interference in fuzzy logic**, fuzzy if-then rules, Fuzzy implications and Fuzzy algorithms, Fuzzifications and Defuzzificataions, Fuzzy Controller, Fuzzy Controllers, Fuzzy Pattern Recognition, Fuzzy Image Processing, Fuzzy Database.
4. **Artificial Neural Network:** Introduction, Artificial Neuron and its model, activation functions, Neural network architecture: single layer and multilayer feed forward networks, re-current networks. Various learning techniques, perception and convergence rule, Auto-associative and hetro-associative memory , Hebb's Learning, Adaline, Perceptron
5. **Multilayer Feed Forward Network**, Back Propagation Algorithms, Different Issues Regarding Convergence of Multilayer Perceptron, Competitive Learning, Self- Organizing, Feature Maps, Adaptive Resonance Theory, Associative Memories, Applications.
6. **Evolutionary and Stochastic Techniques**: Genetic Algorithm (GA), Genetic Representations, (Encoding) Initialization and Selection, Different Operators of GA, Analysis of Selection Operations, Hypothesis of Building Blocks, Schema Theorem and Convergence of Genetic Algorithm, Simulated Annealing and Stochastic Models, Boltzmann Machine, Applications.
7. **Rough Set**: Introduction, Imprecise Categories Approximations and Rough Sets, Reduction of Knowledge, Decision Tables and Applications.
8. **Hybrid Systems**: Neural-Network-Based Fuzzy Systems, Fuzzy Logic-Based Neural Networks, Genetic Algorithm for Neural Network Design and Learning, Fuzzy Logic and Genetic Algorithm for Optimization, Applications

Text Books:

1. Neural Networks, fuzzy Logic and Genetic Algorithm: Synthesis and Applications, S. Rajsekaran and G.A. Vijayalakshmi Pai, Prentice Hall of India.
2. Rough Sets, Z.Pawlak, Kluwer Academic Publisher,1991.
3. Intelligent Hybrid Systems, D. Ruan, Kluwer Academic Publisher,1997

References:

1. Artificial Intelligence and Intelligent Systems, N.P.Padhy, Oxford University Press.
2. Neural Fuzzy Systems, Chin-Teng Lin & C. S. George Lee, Prentice Hall PTR. Addison-Wesley
3. Learning and Soft Computing, V. Kecman, MIT Press,2001
4. Fuzzy Sets and Fuzzy Logic, Klir& Yuan, PHI,1997

IT 324	ELECTIVE-III	ARTIFICIAL INTELLIGENCE
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. To learn about AI problem, Production Systems and their characteristics.
2. To understand the importance of search and the corresponding search strategies for solving AI problem.
3. To introduce to Planning, Natural Language Processing and Expert Systems.

Course Outcomes:

1. The Student understands AI problem characteristics, state space approach for solving AI problem, Production System framework.
2. The student learn several optimal search strategies and the use of heuristics.
3. The student learns relational, inferential, inheritable and procedural knowledge and the corresponding knowledge representation approaches.
4. The student is introduced to applying AI problem solving approaches to natural language processing, planning and expert systems.

Syllabus:

1. **Introduction to Artificial Intelligence:** Artificial Intelligence, AI Problems, AI Techniques, Defining the Problem as a State Space Search, Problem Characteristics, Production Systems.
2. **Search Techniques:** Issues in The Design of Search Programs, Un-Informed Search, BFS, DFS; Heuristic Search Techniques: Generate-And- Test, Hill Climbing, Best- First Search, A^* Algorithm, Problem Reduction, AO^* Algorithm, Constraint Satisfaction, Means-Ends Analysis.
3. **Knowledge Representation using Rules:** Procedural Vs Declarative Knowledge, Logic programming, Forward Vs Backward Reasoning, Matching Techniques, Partial Matching, RETE Matching Algorithm AI Programming languages: Overview of LISP and PROLOG, Production System in Prolog.
4. **Symbolic Logic**: Propositional Logic, First Order Predicate Logic: Representing Instance and is-a Relationships, Computable Functions and Predicates, Unification & Resolution, Natural Deduction;
   Structured Representations of Knowledge: Semantic Nets, Partitioned Semantic Nets, Frames, Conceptual Dependency, Conceptual Graphs, Scripts.
5. **Reasoning under Uncertainty**: Introduction to Non-Monotonic Reasoning, Truth Maintenance Systems, Logics for Non-Monotonic Reasoning, **Statistical Reasoning**: Bayes Theorem, Certainty Factors and Rule-Based Systems, Bayesian Probabilistic Inference, Bayesian Networks, Dempster- Shafer Theory, Fuzzy Logic: Crisp Sets, Fuzzy Sets, Fuzzy Logic Control, Fuzzy Inferences & Fuzzy Systems.
6. **Natural Language Processing**: Steps in The Natural Language Processing, Syntactic Processing and Augmented Transition Nets, Semantic Analysis, NLP Understanding Systems;
   Planning: Components of a Planning System, Goal Stack Planning, Non- linear Planning using Constrait Posting, Hierarchical Planning, Reactive Systems.
7. **Experts Systems:** Overview of an Expert System, Architecture of an Expert Systems, Different Types of Expert Systems- Rule Based, Frame Based, Decision Tree based, Case Based, Neural Network based, Black Board Architectures, Knowledge Acquisition and Validation Techniques, , Knowledge System Building Tools, Expert System Shells.

Text Books:

1. Artificial Intelligence, Elaine Rich and Kevin Knight, Tata Mcgraw-Hill Publications
2. Introduction To Artificial Intelligence & Expert Systems, Patterson, PHI publications

Reference Books:

1. Artificial Intelligence, George F Luger, Pearson Education Publications
2. Artificial Intelligence : A modern Approach, Russell and Norvig, Printice Hall
3. Artificial Intelligence, Robert Schalkoff, Mcgraw-Hill Publications
4. Artificial Intelligence and Machine Learning, Vinod Chandra S.S., Anand HareendranS.

IT 324	ELECTIVE III	IMAGE PROCESSING
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course objectives

1. To explain fundamentals of Image processing concepts.
2. To provide mathematical foundation of image enhancement, image compression and image segmentation.
3. To explain the students about Morphology and its applications in image processing.
4. To explain various methods and techniques for image transformation.

Course outcomes

1. Ability to develop algorithms for fundamental concepts in Image processing.
2. Ability to perform image enhancement, image compression and image segmentation using various methods.
3. Ability to implement Image transformation techniques

Syllabus:

1. **Fundamentals of Image Processing:** Image Acquisition, Image Model, Sampling, Quantization, Relationship between Pixels, Distance Measures, Connectivity, Image Geometry, Photographic Film. Histogram: Definition, Decision Of Contrast Basing On Histogram, Operations Basing on Histograms Like Image Stretching, Image Sliding, Image Classification. Definition and Algorithm of Histogram Equalization.
2. **Image Enhancement in Spatial Domain** : Arithmetic and Logical Operations, Pixel or Point Operations, Size Operations; Smoothing Filters-Mean, Median, Mode Filters – Comparative Study;
3. **Edge enhancement in spatial domain:** Edge enhancement filters, Directorial Filters, Sobel, Laplacian, Robert, KIRSCH Homogeneity & DIFF filters, PREWITT Filter, Contrast based edge enhancement techniques, Comparative study, Low pass filters, High pass filters, Sharpening filters, Comparative study, Color fundamentals and color model
4. **Image Compression:** Run Length Encoding, modified run length encoding, Contour Coding, Huffman Code, Compression Due to Change in Domain, Compression Due to Quantization Compression at the Time of Image Transmission. Brief Discussion on:- Image Compression Standards.
5. **Image Segmentation:** Definition of segmentation, Characteristics of Segmentation, Detection of Discontinuities, Thresholding. Pixel Based Segmentation Method. Region Based Segmentation Methods, Segmentation by Pixel Aggregation, Segmentation by Sub Region Aggregation, Histogram Based Segmentation, Spilt and Merge Technique, Segmentation of moving objects
6. **Morphology:** Dilation, Erosion, Opening, Closing, Hit-And-Miss Transform, Thinning, Thickening, Skeletons , Pruning Extensions to Gray – Scale Images Application of Morphology in I.P
7. **Image Transforms:** A Detail Discussion On Fourier Transform, DFT, FFT, Properties of Fourier transform, WALSH Trans Form, WFT, HADAMARD Transform, DCT Image Enhancement in Frequency Domain: Design of Low Pass, High Pass, EDGE Enhancement, Smoothening Filters in Frequency Domain. Butter Worth Filter, Homomorphic Filters in Frequency Domain Advantages of Filters in Frequency Domain, Comparative Study of Filters in Frequency, Domain and Spatial Domain.

Text Book:

1. Digital Image Processing, Rafael C.Gonzalez And Richard E. Woods, Addision Wesley

Reference Books:

1. Fundamentals Of Electronic Image Processing By Arthyr – R – Weeks,Jr.(PHI)
2. Image Processing, Analysis, And Machine Vision By Milan SonkaVaclanHalavac Roger Boyle, Vikas Publishing House.
3. Digital Image Processing, S. Jayaraman, S. Esakkirajan& T. Veera Kumar, TMH
4. Fundamentals of Digital Image Processing, Chris Solomon, Tobi Breckon, Wiley- Blackwell

IT 324	ELECTIVE III	MOBILE COMPUTING
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Basics of Android**: Introduction to Android Operating System, Version of Android, Installing of software, Android example, Internal Details, Software Stack, Android Core Building Blocks, Android Emulator, AndroidManifest.xml, R.java file, Hide Title Bar, Screen Orientation.
2. **User Interface Widgets**: Working with Button, Toast, Custom Toast, Button, Toggle Button, Switch Button, Image Button, Check Box, Alert Dialog, Spinner, Spinner and other widgets, Auto Complete Text View, Rating Bar, Date Picker. Time Picker, Progress Bar, Activity life cycle and example, Intents-types, Fragment lifecycle and types.
3. **Android Menu**, Layouts and Views: Option Menu, Context Menu, Popup Menu, Types of layouts- Relative, Linear, Table, Grid. Types of views- Grid, Web, Scroll, Search, Tab Host, Dynamic List, Expanded List views.
4. **Android services and Data storage**: web service, Android services, Android Service API, lifecycle and examples. Shared preferences, Soap Vs Restful web service, , Internal storage, External storage, Sqlite Databases, Storing data into external oracle database.
5. **Multimedia and Animation**: Playing audio and video, creating audio player ,Alarm manager, gallery, Animation API, Drawable class, Rotate, Fade, Zoom animations, XML &JSON -XML Parsing SAX, XML Parsing DOM , XML Pull Parser , JSON Parsing.
6. **Speech API and Telephony API**, Web services: Text To Speech API, Example, managing speech and pitch, Speech to text. Telephony manager, Get calls state, call tracker, make phone call and send SMS, Email. Web Service introduction, SOAP vs RESTFUL web services, external oracle data base connections.
7. **Content Providers and Notifications**: Fundamentals of content providers, Content URI, Creation of custom content provider. Notification API, Notification Builder, Issuing notifications, Notification Compact builder, Examples

Text Books:

1. Beginning Android 4 Application Development- WEI-MENG LEE, Wiley India Pvt.ltd

Reference Books:

1. Introduction to Android Application Development: Android Essentials,4/E, Joseph Annuzzi, Jr.Lauren Darcey, Shane Conder, Pearson Education publishers
2. Professional Android 4 Application Development, Reto Meier, Wiley India Pvt.ltd
3. Android Application Development, Pradeep Kothari, Dreamtech publications
http://developer.android.com/guide/index.html

CSE 325	**DATA WARE HOUSING & DATA MINING** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. To understand the evolution of data warehousing and data mining systems
2. To understand extracting, cleaning and transformation of data into a warehouse.
3. To learn the principles of statistics, information theory, machine learning and other areas AI and implementation of data mining techniques.
4. To understand pattern mining using classification and clustering methods.

Course Outcomes:

1. The student understands the differences between OLTP and OLAP.
2. The student learns how data cube technology supports summarization and querying high dimensional data.
3. The student is introduced to similarity, distance, information gain and other performance and error metrics used for evaluation of mining results.
4. The student is introduced to various approaches to association rule mining, supervised and unsupervised learning and the corresponding classification and clustering approaches involving decision trees, Bayesian approaches, model based and agglomerative approaches.

Syllabus:

1. **Introduction to Data Mining:** Importance of Data Ware housing and Data Mining, Kinds of Patterns, Technologies, Applications, Major Issues in Data Mining, Data Objects and Attributes Types, Statistical Descriptions of Data, Estimating Data Similarity and Dissimilarity.
2. **Data exploration and pre-processing:** Data Visualization, Quality data**,** Data Cleaning, Data Integration, Data Reduction, Data Transformation, Discretization and Concept Hierarchy Generation.
3. **Data Warehouse and OLAP Technology:** Basic Concepts of Data warehouse, Data Modeling using Cubes and OLAP, DWH Design and usage, Implementation using Data Cubes and OLAPs, Data Generalization with AOI.
4. **Data Cube Technology:** Preliminary Concepts of Data Cube Computation, Data Cube Computation Methods: Multi-way Array Aggregation for Full Cube, BUC, Star-cubing, Pre-computing shell fragments for High dimensional OLAP.
5. **Mining Frequent Patterns Based on Associations and Correlations**: Basic Concepts, Frequent Item set Mining Methods: Apriori Algorithm, Association Rule Generation, Improvements to A Priori , FP- Growth Approach, Pattern Evaluation Method**s.**
6. **Classification & Prediction**: Basic Concepts, Decision Tree Induction, Bayes Classification, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy, Classification by Back Propagation, K-nearest neighbor classifier.
7. **Cluster Analysis:** Basic Concepts and issues in clustering, Types of Data in Cluster Analysis, Partitioning Methods, Hierarchical Methods, DBSCAN, Grid Based Methods, Evaluation of Clustering Solutions.

Text Books:

1. Data Mining- Concepts and Techniques by Jiawei Han, Micheline Kamber and Jian Pei–Morgan Kaufmann publishers –--3rd edition
2. Data Mining Techniques, A.K.Pujari, University Press

Reference Books:

1. Data mining concepts byTan, Steinbech, and Vipin Kumar - Pearson Edu publishers

IT 327	NETWORK PROGRAMMING LAB	
Instruction:3Periods/week,	Univ. Exam: 3Hours	**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

FIRST CYCLE OF EXPERIMENTS

1. PC-to-PC COMMUNICATIONS UNDER DO SWITH NULL MODEM
   a) Using Serial Ports and RS-232C Cable Connection
   b) Using Parallel Ports and Parallel Cable Connection
2. PC-to-PC COMMUNICATIONS UNDER DO SWITH MODE Mand4-LINE EXCHANGE Using Communication Software: COMIT or XTALK
3. PC-to-PC COMMUNICATIONS UNDER WIN98's DIRECT CABLE CONNECTION with NULL MODEM
   a) Using Serial Ports and RS-232C Cable Connection
   b) Using Parallel Ports and Parallel Cable Connection
4. PC-to-PC COMMUNICATIONS UNDER WIN98's DIAL-UP NETWORKING WITH MODEM and 4-LINE EXCHANGE.
5. PC-to-PC COMMUNICATIONS UNDER WIN98's HYPER TERMINAL WITH MODEM and 4-LINE EXCHANGE.
6. a) LAN WITH BUS TOPOLOGY with a minimum of two systems
      j) Windows Peer-to-Peer Network
      ii) Windows NT Client- Server Network
   b) LAN WITH STAR TOPOLOGY with a minimum of two systems
7. a) LAN WITH BUS TOPOLOGY with a minimum of two systems using NOVELL Netware
   b) LAN WITH STAR TOPOLOGY with a minimum of two systems using NOVELL Netware

SECOND CYCLE OF EXPERIMENTS

1. INERNET CONNECTION SET-UP USING DIAL-UP NETWORKING
2. TERMINAL NETWORK WITH UNIX/LINUX SERVER and one or two Terminals
3. TERMINAL NETWORK WITH UNIX/LINUX SERVER, Terminal Server, and one or two terminals
4. NETWORK PROGRAMMING EXERCISE-I: USING A SIMPLIFIED API Echo software (Develop echo client and echo server programs and run the two programs on separate computers and verify that they can communicate Chat software (Develop chat client and chat server programs and test to ensure they can communicate). Build a simple file transfer service that consists of client and server.
5. NETWORK PROGRAMMING EXERCISE-II: USING THE SOCKET API Write an echo client and server using sockets build a web server using sockets.
6. CONCURRENT NETWORK PROGRAMMING EXERCISE–III: Build a Concurrent server (threads)– Create a server capable of handling connections from multiple clients concurrently Build a Concurrent file transfer server (processes)– Create separate processes to allow a server to handle multiple clients concurrently.
7. NETWORK PROGRAMMING EXERCISE–IV: USING PROTOCOL DESIGN Design a reliable data transfer protocol (Devise, implement and test a protocol that provides reliable data transfer across a network that drops, delays or corrupts packets Design stop and wait flow control protocol Design a sliding window protocol
8. NETWORK PROGRAMMING EXERCISE–V: WITH PROTOCOLS FROM TCP/IP SUITE Build a domain name system client program

CSE328	**WEB TECHNOLOGIES LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits:1.5**
Internal:50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Each student should develop two projects out of this list using JSP, JDBC, J2EE

1. Design Airlines Ticket Reservation System
2. Design ONLINE Banking system.
3. Design Library Information system
4. Design Gram Panchayat Information system for House tax, water tax, wealth tax, Library tax collection, phone bill, Electricity bill collection.
5. Design student information system portal which maintain attendance, marks etc.
6. Design online examination system.

B.TECH (IT) 4th YEAR I-SEMESTER SCHEME OF INSTRUCTION AND EXAMINATION
With effect from 2019-2020 admitted batch

Sub Ref. No	Name of the Subject	Periods			Maximum marks			Credits
		Theory	Tutorial	Lab	Exam	Internal	Total	
CSE411	ELECTIVE-IV	3	--	--	70	30	100	3
CSE412	ELECTIVE-V	3	--	--	70	30	100	3
CSE413	GPS APPLICATIONS	3	--	--	70	30	100	3
IT414	BIO-INFORMATICS	3	--	--	70	30	100	3
IT415	PROJECT-II	--	--	12	50	50	100	6
CSE416	ELECTVE-IV LAB	--	--	3	50	50	100	1.5
CSE417	ELECTIVE-V LAB	--	--	3	50	50	100	1.5
	TOTAL	12	0	18	430	270	700	**21**

ELECTIVE-IV
1. Big Data Analytics
2. Machine Learning
3. R Programming

ELECTIVE-V
1. Cryptography & Network Security
2. IOT
3. Cloud Computing

CSE 411	ELECTIVE-IV **BIGDATA ANALYTICS** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

- To introduce Big Data and the Data analytics lifecycle to address business challenges that leverage big data.
- To understand the importance of mining data streams and social network graphs.
- To introduce big data analytics technology and tools including MapReduce and Hadoop.

Expected Course Outcome:

- Reframe a business challenge as an analytics challenge.
- Apply appropriate analytic techniques and tools to analyze big data.
- Create models and identify insights that can lead to actionable results.
- Effectively participate in big data and other analytics projects.
- Use tools such as Map Reduce / Hadoop.

Syllabus:

1. **Big Data Concepts and Environment**: Big Data Overview-Big Data Challenges and Opportunities- Data analytics lifecycle overview – Phases of Data Analytics: Discovery, Data preparation, Model planning, Model building.
2. **Overview of Hadoop and HDFS**: Introduction to Hadoop - The Distributed File System: HDFS, GPFS – The Design of HDFS – HDFS-Concepts-Blocks, Name Nodes and Data Nodes; Components of Hadoop-Hadoop Cluster Architecture-Batch Processing- Serialization - Hadoop ecosystem of tools-NoSQL .
3. **Map Reduce**: MapReduce Basics - Functional Programming Roots - Mappers and Reducers - The Execution Framework -MapReduce Algorithm Design –Shuffling, Grouping, Sorting- Custom Partitioners and Combiners- MapReduce Formats and Features.
4. **Data Stream Mining**: The Stream Data Model: A Data-Stream-Management System, Examples of Stream Sources, Stream Queries, Issues in Stream Processing. Sampling Data in a Stream, Filtering Streams (The Bloom Filter), Counting Distinct Elements in a Stream, Counting Ones in a Window, Decaying Windows.
5. **Big Data Clustering:** Overview of clustering techniques, Hierarchical Clustering, Partitioning Methods, CURE algorithm, Clustering stream
6. **Mining Social Network Graphs**: Link Analysis: Page Rank- Efficient computation of Page Rank- Topic Sensitive Page Rank- Link Spam- Hubs and Authorities. Mining Social Network Graphs: Web Advertising: Online and Offline Algorithms; Social Network Graphs: Clustering of Social Network Graphs- Direct Discovery of Communities- Partitioning of Graphs- Finding overlapping communities- Sim rank Counting Triangles- Neighborhood properties of Graphs.

Text Book:

1. RadhaShankarmani, M Vijayalakshmi, “Big Data Analytics”, 2nd Edition, Wiley

Reference Books:

1. AnandRajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2014.
2. Data Warehousing in the Age of Big Data, Krish Krishnan Elsevier Inc, 2013

CSE 411	ELECTIVE-IV **MACHINE LEARNING** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction to Machine Learning**, Applications of Machine learning, Supervisory Learning: Learning classes from examples, Vapnik- Charvonenkis (VC) Dimension, Probably Approximately Correct(PAC) Learning, noise, learning multiple classes, regression, model selection and generalization, dimensions of supervised machine learning algorithms
2. **Bayesian Decision Theory**: Classification, losses and risks, discriminant functions, utility theory, value of information, Bayesian networks, Influence diagrams, Association rules, Parametric Methods: Maximum likelihood estimation, evaluating an estimator with bias and variance, Bayes' estimator, parametric classification, regression, tuning model complexity: bias vs variance dilemma, model selection procedures
3. **Multivariate methods**: Multivariate data, parameter estimation, missing value imputation, univariate normal distribution and classification, discrete features, regression, Dimensionality Reduction: Subset selection, PCA, Factor Analysis, multi-dimensional scaling, LDA
4. **Clustering**: Mixture densities, K-means clustering, Expectation Maximization algorithm, mixtures of Latent Variable Models, Supervised learning after clustering, Hierarchical clustering, choosing number of clusters
5. **Non-parametric methods**: Non-parametric methods density estimation, generalisation to multivariate data, nonparametric classification, condensed nearest neighbors, non-parametric regression: smoothing models, choosing smoothing parameters
6. **Decision trees and Linear Discrimination**: Univariate classification and regression trees, rule extraction from trees, Multivariate trees, Generalizing linear model, two class and multi-class geometry of linear discriminant, pairwise separation, gradient descent, logistic discrimination for binary and multi-class problems, Support vector machines, optimal separating hyperplane, kernel functions for non-separable spaces, SVM for regression.
7. **Hidden Markov Models**: Discrete Markov processes, Hidden Markov Models, Three basic problems of HMM, Evaluation problem, finding the state sequence, Learning model parameters, continuous observations, Model selection in HMM Assessing and comparing classification Algorithms: Cross-validation and resampling methods, measuring error, interval estimation, hypothesis testing, assessing performance of a classifier, comparing two classification algorithms, comparing multiple classification algorithms based on variance

Text Book:

1. Introduction to Machine Learning by Ethem Alpaydin, Prentice-Hall of India, 2006
2. Machine Learning by Saikat Dutt and Subramanian Chandramouli

Reference books:

1. Machine Learning, Tom Mitchell , McGraw Hill, 1997
2. Pattern Classification, Richard 0. Duda, Peter E. Hart and David G. Stork, John Wiley & Sons Inc., 2001

CSE 411	**ELECTIVE-IV R PROGRAMMING** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3 Periods +0Tut/week,	Univ. Exam: 3Hours	**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

After taking the course, students will be able to

1. Use R for statistical programming, computation, graphics, and modeling,
2. Write functions and use R in an efficient way,
3. Fit some basic types of statistical models
4. Use R in their own research,
5. Be able to expand their knowledge of R on their own.

Course Outcomes:

At the end of this course, students will be able to:

1. List motivation for learning a programming language
2. Access online resources for R and import new function packages into the R workspace
3. Import, review, manipulate and summarize data-sets in R
4. Explore data-sets to create testable hypotheses and identify appropriate statistical tests
5. Perform appropriate statistical tests using R Create and edit visualizations

Syllabus:

1. **Introduction**, How to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes**.**
2. **R Programming Structures**, Control Statements, Loops, - Looping Over Non vector Sets,- If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values, Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective, No Pointers in R, Recursion, A Quick sort Implementation-Extended Extended Example: A Binary Search Tree.
3. **Doing Math and Simulation in R**, Math Function, Extended Example Calculating Probability- Cumulative Sums and Products-Minima and Maxima- Calculus, Functions Fir Statistical Distribution, Sorting, Linear Algebra Operation on Vectors and Matrices, Extended Example: Vector cross Product- Extended Example: Finding Stationary Distribution of Markov Chains, Set Operation, Input /output, Accessing the Keyboard and Monitor, Reading and writer Files,
4. **Graphics,** Creating Graphs, The Workhorse of R Base Graphics, the plot () Function – Customizing Graphs, Saving Graphs to Files.
5. **Probability Distributions**, Normal Distribution- Binomial Distribution- Poisson Distributions Other Distribution, Basic Statistics, Correlation and Covariance, T-Tests,-ANOVA.
6. **Linear Models**, Simple Linear Regression, -Multiple Regression Generalized Linear Models, Logistic Regression, - Poisson Regression- other Generalized Linear Models-Survival Analysis, Nonlinear Models, Splines- Decision- Random Forests,

Text Books:

1) The Art of R Programming, A K Verma, Cengage Learning.
2) R for Everyone, Lander, Pearson
3) The Art of R Programming, Norman Matloff, No starch Press.

Reference Books:

1) R Cookbook, Paul Teetor, Oreilly.
2) R in Action, Rob Kabacoff, Manning

CSE 412	ELECTIVE-V **CRYPTOGRAPHY &NETWORK SECURITY** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. Introduction of the issues in network security- its need and importance, taxonomy and terminology.
2. Discussion of various cryptographic techniques.
3. Exploration of different types of security threats and remedies.
4. Understanding of Internet security protocols and standards

Course Outcomes:

1. Realize the need and importance of network and data security in the Internet and in the distributed environments.
2. Identify the different types of network security issues and their remedies.
3. Application various cryptographic tools and techniques in different contexts and as per need of security levels.
4. Implementation of some Internet security protocols and standards

Syllabus:

1 **Overview:** Computer Security Concepts, Threats, Attacks, and Assets, Security Functional Requirements, A Security Architecture for Open Systems, Computer Security Trends, Computer Security Strategy. Cryptographic Tools: Confidentiality with Symmetric Encryption, Message Authentication and Hash Functions, Public-Key Encryption, Digital Signatures and Key Management, Random and Pseudorandom Numbers, Practical Application: Encryption of Stored Data. User Authentication: Means of Authentication, Password-Based Authentication, Token-Based Authentication, Biometric Authentication, Remote User Authentication, Security Issues for User Authentication, Practical Application: An Iris Biometric System, Case Study: Security Problems for ATM Systems.

2 **Access Control:** Access Control Principles, Subjects, Objects, and Access Rights, Discretionary Access Control, Example: UNIX File Access Control, Role-Based Access Control, Case Study: RBAC System for a Bank. Database Security: The Need for Database Security, Database Management Systems, Relational Databases, Database Access Control, Inference, Statistical Databases, Database Encryption, Cloud Security.

3 **Malicious Software:** Types of Malicious Software (Malware), Propagation—Infected Content—Viruses, Propagation—Vulnerability Exploit—Worms, Propagation—Social Engineering—SPAM E-mail, Trojans, Payload—System Corruption, Payload—Attack Agent—Zombie, Bots, Payload—Information Theft—Key loggers, Phishing, Spyware, Payload—Steal thing—Backdoors, Root kits, Countermeasures.
Denial-of-Service Attacks: Denial-of-Service Attacks, Flooding Attacks, Distributed Denial-of-Service Attacks, Application-Based Bandwidth Attacks, Reflector and Amplifier Attacks, Defenses Against Denial-of-Service Attacks, Responding to a Denial-of-Service Attack.

4 **Intrusion Detection:** Intruders, Intrusion Detection, Host-Based Intrusion Detection, Distributed Host-Based Intrusion Detection, Network-Based Intrusion Detection, Distributed Adaptive Intrusion Detection, Intrusion Detection Exchange Format, Honey pots, Example System: Snort. Firewalls and Intrusion Prevention Systems: The Need for Firewalls, Firewall Characteristics, Types of Firewalls, Firewall Basing, Firewall Location and Configurations, Intrusion Prevention Systems, Example: Unified Threat Management Products.

5 **Buffer Overflow:** Stack Overflows, Defending Against Buffer Overflows, Other Forms of Overflow Attacks, Software Security: Software Security Issues, Handling Program Input, Writing Safe Program Code, Interacting with the Operating System and Other Programs, Handling Program Output. Operating System Security: Introduction to Operating System Security, System Security Planning, Operating Systems Hardening, Application Security, Security Maintenance, Linux/Unix Security, Windows Security, Virtualization Security.

6 **Symmetric Encryption and Message Confidentiality:** Symmetric Encryption Principles, Data Encryption Standard, Advanced Encryption Standard, Stream Ciphers andRC4, Cipher Block Modes of Operation, Location of Symmetric Encryption Devices, Key Distribution. Public-Key Cryptography and

Message Authentication: Secure Hash Function, HMAC, The RSA Public-Key Encryption Algorithm, Diffie-Hellman and Other Asymmetric Algorithms.

7 **Internet Security Protocols and Standards:** Secure E-mail and S/MIME, Domain Keys Identified Mail, Secure Socket Layer (SSL) and Transport Layer Security (TLS), HTTPS, IPv4 and IPv6 Security. Internet Authentication Applications: Kerberos, X.509, Public-Key Infrastructure, Federated Identity Management. Wireless Network Security: Wireless Security Overview, IEEE 802.11 Wireless LAN Overview, IEEE 802.11i Wireless LAN Security.

Text Book:

1. Computer Security - Principles and Practices (Except the Chapters 13, 14, 15, 16, 17,18, 19), 2nd Edition by William Stallings, Pearson Education, Inc.

Reference Books:

1. Cryptography and Network Security by William Stallings, Pearson Education Asia, New Delhi.
2. Network Security Essentials Applications and Standards, by William Stallings, Pearson Education Asia, New Delhi.

CSE 412	ELECTIVE-V **INTERNET OF THINGS** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1) **Introduction to the internet of things**. IoT Architecture: History of IoT, M2M –Machine to Machine, Web of Things, IoT protocols The Architecture The Layering concepts, IoT Communication Pattern, IoT protocol Architecture, The 6LoWPAN
2) **Prototyping connected objects**. Open-source prototyping platforms, Basics of IoT Networking(IOT components and Gateways)
3) **Integrating internet services**. XML and JSON. HTTP APIs for accessing popular Internet services (Facebook, Twitter, and others). Practical activities. IoT Application Development: Application Protocols MQTT, REST/HTTP, CoAP, MySQL, RPL,
4) **Overview of IoT** supported Hardware platforms such as: Raspberry pi, ARM Cortex Processors, Arduino and Intel Galileo boards
5) **Ubiquitous computing**, applications of IOT, Virtualization of network resources and physical devices in IOT.
6) **Internet of Things** Standardisation M2M Service Layer Standardisation OGC Sensor Web for IoT.

Text Books:

1. Internet Of Things: Converging Technologies For Smart Environments And Integrated Ecosystems, Marina Ruggieri H, River Publishers Series In Communications.

CSE 412	ELECTIVE-V **CLOUD COMPUTING** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

1. To import fundamental concepts in the area of cloud computing.
2. To understand the concept of Virtualization and cloud data storage.
3. To learn cloud Application Development and cloud Governance.
4. To gain competence in Map Reduce and Hadoop Overview.

Course Outcomes:

1. Identify the architecture and infrastructure of cloud computing.
2. Develop applications for cloud computing.
3. Design and Implement a novel cloud computing application.

Syllabus:

1. **Introduction to cloud computing:** Cloud computing components, Infrastructure services, storage applications, database services – introduction to Saas, Paas, Iaas, Idaas, data storage incloud
2. **Virtualization:** enabling technologies, types of virtualization, server virtualization, desktop virtualization, memory virtualization, application and storage virtualization-tools and products available for virtualization
3. **SAAS and PAAS:** Getting started with Saas, SaaS solutions, SOA, PaaS and benefits.
4. **Iaas and Cloud data storage:** understanding Iaas, improving performance for load balancing, server types within Iaas, utilizing cloud based NAS devices, cloud based data storage, and backup services, cloud based block storage and database services
5. **Cloud Application development:** Client server distributed architecture for cloud designing cloud based solutions, coding cloud based applications, traditional Apps vs cloud Apps, client side programming, server side programming overview-fundamental treatment of web application frameworks.
6. **Cloud Governance and economics:** Securing the cloud, disaster recovery and business continuity in the cloud, Managing the cloud, migrating to the cloud, governing and evaluating the clouds business impact and economics,
7. **Inside Cloud**: Introduction to MapReduce and Hadoop-over view of big data and its impact on cloud

Text Books:

1. Cloud Computing: SaaS, PaaS, IaaS, Virtualization, Business Models, Mobile, Security and More, Kris Jamsa, Jones & Bartlett Publishers, Paper back edition,2013
2. Hadoop Map Reduce cookbook, Srinath Perera and Thilina Gunarathne, Packt publishing

Reference Book:

1. Cloud Computing: A Practical Approach, Anthony T .Velte, Toby J.Velte, Robert Elsenpeter, Tata McGraw Hill Edition

CSE 413	**GPS APPLICATIONS** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. Development of NAVSTAR GPS. GPS Satellite configuration- Space segment, Control segment, User segment.
2. GPS working principle, basic equations for finding user position, user position determination with least squares estimator.
3. Other Global Satellite Constellations, GLONASS, GALILEO, Comparison of 3 GNSS (GPS, GALILEO, GLONASS) in terms of constellation and services provided.
4. GPS Signal generation, Pseudorandom noise (PRN) code, C/A code , P code, Navigation data, Signal structure of GPS, signal power.
5. Coordinate Systems: Geoid, Ellipsoid, Coordinate Systems, Geodetic and Geo centric coordinate systems, ECEF coordinates, world geodetic 1984 system, Conversion between Cartesian and geodetic coordinate frame.
6. GPS Error sources, ionospheric effects on GPS signals and its mitigation methods.
7. Satellite based augmentation system-need for GPS augmentation, GPS Aided GEO Augmented System (GAGAN).

Text books:

1. G S RAO, Global Navigation Satellite Systems, McGraw-Hill Publications, New Delhi, 2010
2. Pratap Mishra, Global positioning system: signals, measurements, and performance, Ganga-Jamuna Press, 2006.

Reference Books:

1. Scott Gleason and Demoz Gebre- Egziabher, GNSS Applications and Methods, Artech House, 685 Canton Street, Norwood, MA 02062, 2009.
2. James Ba – Yen Tsui, 'Fundamentals of GPS receivers – A software approach', John Wiley & Sons (2001).
3. B.Hoffmann- Wellenhof, GPS theory and practice, 5th Edition, Springer 2001.

IT 414	BIOINFORMATICS	
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction:** Definitions, Sequencing, Biological sequence/structure, Genome Projects, Pattern recognition an prediction, Folding problem, Sequence Analysis, Homology and Analogy.
2. **Protein Information Resources:** Biological databases, Primary sequence databases, Protein Sequence databases, Secondary databases, Protein pattern databases, and Structure classification databases.
3. **Genome Information Resources :**DNA sequence databases, specialized genomic resources
4. **DNA Sequence analysis :**Importance of DNA analysis, Gene structure and DNA sequences, Features of DNA sequence analysis, EST (Expressed Sequence Tag) searches, Gene hunting, Profile of a cell, EST analysis, Effects of EST data on DNA databases
5. **Pair wise alignment techniques :** Database searching, Alphabets and complexity, Algorithm and programs, Comparing two sequences, sub-sequences, Identity and similarity, The Dot plot, Local and global similarity, different alignment techniques, Dynamic Programming, Pair wise database searching.
6. **Multiple sequence alignment :** Definition and Goal, The consensus, computational complexity, Manual methods, Simultaneous methods, Progressive methods, Databases of Multiple alignments and searching
7. **Secondary database searching:** Importance and need of secondary database searches, secondary database structure and building a sequence search protocol
8. **Analysis packages:** Analysis package structure, commercial databases, commercial software, comprehensive packages, packages specializing in DNA analysis, Intranet Packages, Internet Packages.

Text Books:

1. Introduction to Bioinformatics, T K Attwood & D J Parry-Smith Addison Wesley Longman
2. Bioinformatics- A Beginner's Guide, Jean-Michel Claveriw, Cerdric Notredame WILEY dreamlech India Pvt. Ltd

Reference Books:

1. Introduction to Bioinformatics, Arthur M.Lesk, OXFORD publishers (Indian Edition)

CSE 416	ELECTIVE-IV LAB **BIG DATA ANALYTICS LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Getting Hadoop Up and Running in a cluster:

1. Setting up Hadoop on standalone machine.
2. Word count Map Reduce program using standalone Hadoop.
3. Adding the combiner step to the Word count Map Reduceprogram.
4. Setting up HDFS.
5. Using HDFS monitoring UI
6. HDFS basic command-line file operations.
7. Setting Hadoop in a distributed cluster environment.
8. Running the Word Count program in a distributed cluster environment.
9. Using Map Reduce monitoring UI

Hadoop Map Reduce Applications:

10. Choosing appropriate Hadoop data types.
11. Implementing a custom Hadoop Writable data type.
12. Implementing a custom Hadoop key type.
13. Emitting data of different value types from a mapper.
14. Choosing a suitable Hadoop Input Format for your input data format.
15. Formatting the results of Map Reduce Computation – using Hadoop Output Formats.

Analytics

1. Simple analytics using Map Reduce.
2. Performing Group-By using Map Reduce.
3. Calculating frequency distributions and sorting using Map Reduce.
4. Plotting the Hadoop results using GNU plot.
5. Calculating histograms using Map Reduce.
6. Calculating scatter plots using Map Reduce.
7. Parsing a Complex dataset with Hadoop.
8. Joining two datasets using Map Reduce.

Text Book:

1. Hadoop Map Reduce Cookbook, Srinath Perera & Thilina Gunarathne, 2013, PACKT PUBLISHING.

CSE 416	ELECTIVE-IV LAB **MACHINE LEARNING LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3 Periods +0Tut/week,	Univ.Exam: 3Hours	**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Course objectives:

This course will enable students to

1. Make use of Data sets in implementing the machine learning algorithms.
2. Implement the machine learning concepts and algorithms in any suitable language of choice.

Course outcomes:

The students should be able to:

1. Understand the implementation procedures for the machine learning algorithms.
2. Design Java/Python programs for various Learning algorithms.
3. Apply appropriate data sets to the Machine Learning algorithms.
4. Identify and apply Machine Learning algorithms to solve real world problems.

Description (If any):

1. The programs can be implemented in either JAVA or Python.
2. For Problems 1 to 6 and 10, programs are to be developed without using the built-in classes or APIs of Java/Python.
3. Data sets can be taken from standard repositories (https://archive.ics.uci.edu/ml/datasets.html) or constructed by the students.

Lab Experiments:

1. Implement and demonstrate the **FIND-S algorithm** for finding the most specific hypothesis based on a given set of training data samples. Read the training data froma.CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the **Candidate-Elimination algorithm** to output a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree based **ID3 algorithm**. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4. Build an Artificial Neural Network by implementing the **Back propagation algorithm** and test the same using appropriate data sets.
5. Write a program to implement the **naïve Bayesian classifier** for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Assuming a set of documents that need to be classified, use the **naïve Bayesian Classifier** model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7. Write a program to construct a **Bayesian network** considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply **EM algorithm** to cluster a set of data stored in a .CSV file. Use the same data set for clustering using ***k*-Means algorithm**. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement ***k*-Nearest Neighbor algorithm** to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric **Locally Weighted Regression algorithm** in order to fit data points. Select appropriate data set for your experiment and draw graphs.

CSE 416	ELECTIVE-IV LAB **R- PROGRAMMIMG LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3 Periods +0Tut/week, Univ. Exam: 3Hours		**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Lab Experiments:

1. Implement the Following
   a. To create a data frame df1 to contain 10 observations and 3 variables, column with letters, column 2 with random numbers and column 3 with first 10 natural numbers.
   b. Create df3 by merging df1 by column1 with another data frame df2 containing 20 observations and 2 variables column4 with letters, column5 with sequence of 20 real numbers from 0 to 1 in equal steps
   c. Find the dimensionality of data framedf3.
   d. Rename observations whose column1 value is 'D' from data framedf3
2. Implement the following
   a. Create h1 to contain 1000 random numbers, distributed in normal distribution and plot the histogram with colors.
   b. Create a data frame to contain randomly drawn samples of 25 cards from 52 distinct cards with replacements. Use 'table' function to find the 'duplicated' and tabulate the list of cards and their frequency of occurrence in the sample.
3. Write R Program using 'apply' group of functions to create and apply normalization function on each of the numeric variables/columns of iris dataset to transform them into
   a. 0 to 1 range with min-max normalization.
   b. a value around 0 with z-score normalization.
4. Create a data frame with 10 observations and 3 variables and add new rows and columns to it using 'rbind' and 'cbind' function.
5. Create a function to discretize a numeric variable into 3 quantiles and label them as low, medium, and high. Apply it on each attribute of iris dataset to create a new data frame. 'discrete_ iris' with Categorical variables and the class label.
6. Write R program to find the approximate value of π (pi) by simulation using a large number of uniformly distributed data points with their coordinates in the range of [-1,1] and find the ratio of number of points within the circle of radius 1, to total number of data points. Observe the improvement in accuracy of result with the increased number of data points distributed.
7. Write R programs to find the probability of a variable to have a given value in different distributions like Uniform, Normal, Poisson and Binomial using 'pnorm', 'ppois', and the other such functions.
8. Apply 'ddply' for data summarization of iris dataset based on 'species' and get the same summarization using 'sqldf'
9. After attaching data set 'mtcars' to access its variables, use R statements to visulalize the relationship between the variables of 'mtcars':
   a. Using scatter plots with colors.
   b. Box plots showing the spread of the variable 'mpg' for different values of'cyl'.
   c. Find correlations between all pairs of variables.
10. Write R program to implement linear and multiple regression on 'mtcars' dataset to estimate the value of 'mpg' variable, with best R^2 and plot the original values in 'green' and predicted values in 'red'.
11. Write R program to create new variables in low dimensional space using
    a. PCA and
    b. SVD and use them for predicting the values of 'mpg' variable.
12. Write R Programs to apply k-mean clustering on 'iris' data set and get the summary statistics. Implement a mini-project to process a collection of text documents / tweets and apply tokenization, stop word removal and stemming to represent the collection as a document – term matrix reflecting the term frequencies. Cluster the documents using a simple clustering algorithm and estimate the purity of the clustering solution.

CSE417	ELECTIVE-V LAB **CRYPTOGRAPHY & NETWORK SECURITY LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3 Periods/week	Univ. Exam: 3Hours	**Credits:1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Lab Experiments:

1. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a Java program to perform encryption and decryption using the following algorithms
   a. Ceaser cipher
   b. Substitution cipher
   c. Hill Cipher
4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA.

CSE 417	**ELECTIVE-V LAB: INTERNET OF THINGS LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods Univ. Exam: 3 Hours		**Credits: 1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

List of Experiments:

1. Study Of Various Network Protocols Used In IOT.
2. Application of WIFI In IOT Systems.
3. Application of 6lowpan In IOT Systems.
4. Application of Bluetooth In IOT Systems.
5. Application of 802.15.4 Zigbee. In IOT Systems.
6. Design a Simple IOT System Comprising Sensors, Wireless Network Connection, Data Analytics

CSE417	**ELECTIVE-V LAB : CLOUD COMPUTING LAB** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3 Periods/week		**Credits:1.5**
Internal: 50 Marks	**University Exam: 50 Marks**	**Total: 100 Marks**

Use Eucalyptus or Open Nebula or equivalent to set up the cloud and demonstrate.

1. Find procedure to run the virtual machine of different configuration. Check how many virtual machines can be utilized at particular time.
2. Find procedure to attach virtual block to the virtual machine and check whether it holds the data even after the release of the virtual machine.
3. Install a C compiler in the virtual machine and execute a sample program.
4. Show the virtual machine migration based on the certain condition from one node to the other.
5. Find procedure to install storage controller and interact with it.

B.TECH (IT) 4th YEAR II-SEMESTER SCHEME OF INSTRUCTION AND EXAMINATION
With effect from 2019-2020 admitted batch

Sub Ref. No	Name of the Subject	Periods			Maximum marks			Credits
		Theory	Tutorial	Lab	Exam	Internal	Total	
IT421	ELECIVE-VI	3	--	--	70	30	100	3
CSE422	ENTREPRENEURSHIP	3	--	--	70	30	100	3
CSE423	1G-4G MOBILE COMMUNICATION NETWORKS	3	--	--	70	30	100	3
IT424	PROJECT-III	--	--	12	50	50	100	6
	TOTAL	9	0	12	260	140	400	**15**

ELECTIVE-VI

1. Cyber Security & Digital Forensics
2. Software Project Management
3. E-Commerce

IT421	ELECTVE-VI CYBER SECURITY AND DIGITAL FORENSICS	
Instruction: 3 Periods & 0 Tut/week		**Credits:3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction to Information Security Fundamentals and Best Practices**
   - Protecting Your Computer and its Contents
   - Securing Computer Networks--Basics of Networking
   - Compromised Computers
   - Secure Communications and Information Security Best Practices
   - Privacy Guidelines
   - Safe Internet Usage
2. **Ethics in Cyber Security & Cyber Law**
   - Privacy
   - Intellectual Property
   - Professional Ethics
   - Freedom of Speech
   - Fair User and Ethical Hacking
   - Trademarks
   - Internet Fraud
   - Electronic Evidence
   - Cybercrimes
3. **Penetration Testing**
   - Overview of the web from a penetration testers perspective
   - Exploring the various servers and clients
   - Discussion of the various web architectures
   - Discussion of the different types of vulnerabilities
   - Defining a web application test scope and process
   - Defining types of penetration testing
4. **Web Application Security**

   Common Issues in Web Apps

   What is XSS, SQL injection, CSRF, Password Vulnerabilities, SSL, CAPTCHA, Session Hijacking, Local and Remote File Inclusion, Audit Trails, Web Server Issues
5. **Forensics & Network Assurance**
   - Forensic Technologies
   - Digital Evidence Collection
   - Evidentiary Reporting
   - Layered Defense
   - Surveillance and Reconnaissance
   - Outsider Thread Protection
6. **Information Risk Management**
   - Asset Evaluation and Business Impact Analysis
   - Risk Identification
   - Risk Quantification
   - Risk Response Development and Control
   - Security Policy, Compliance, and Business Continuity
   - Forensic investigation using Access Data FTK, En-Case
7. **Cyber Incident Analysis and Response**
   - Incident Preparation
   - Incident Detection and Analysis
   - Containment, Eradication, and Recovery
   - Proactive and Post-Incident Cyber Services
   - CIA triangle

Reference Books:

1. The Official CHFI Study Guide for Computer Hacking Forensic Investigator by Dave Kleiman
2. CISSP Study Guide, 6th Edition by James M.Stewart
3. www.**nist**.gov/

IT421	ELECTIVE-VI	SOFTWARE PROJECT MANAGEMENT
Instruction: 3 Periods & 0 Tut/week Univ. Exam: 3 Hours		**Credits:3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Course Objectives:

The main goal of software development projects is to create a software system with a predetermined functionality and quality in a given time frame and with given costs. For achieving this goal, models are required for determining target values and for continuously controlling these values. This course focuses on principles, techniques, methods &tools for model-based management of software projects, assurance of product quality and process adherence (quality assurance), as well as experience-based creation & improvement of models (process management).

The goals of the course can be characterized as follows:

1. Understanding the specific roles within a software organization as related to project and process management
2. Understanding the basic infrastructure competences (e.g., process modeling and measurement)
3. Understanding the basic steps of project planning, project management, quality assurance, and process management and their relationships.

Syllabus:

1. **Conventional Software Management**: The waterfall model, conventional software Management performance. Evolution of Software Economics: Software Economics, pragmatic software cost estimation.
2. **Improving Software Economics**: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections. The old way and the new: The principles of conventional software engineering, principles of modern software management, transitioning to an iterative process.
3. **Life cycle phases**: Engineering and production stages, inception, Elaboration, construction, transition phases.
   Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts. Model based software architectures: A Management perspective and technical perspective.
4. **Work Flows of the process**: Software process workflows, Inter Trans workflows, Check points of the Process: Major Mile Stones, Minor Mile stones, Periodic status assessments. Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Interaction planning process, Pragmatic planning.
5. **Project Organizations and Responsibilities**: Line-of-Business Organizations, Project Organizations, evolution of Organizations.
   Process Automation: Automation Building Blocks, the Project Environment.
6. **Project Control and Process instrumentation**: The server care Metrics, Management indicators, quality indicators, life cycle expectations pragmatic Software Metrics, Metrics automation. Tailoring the Process: Process discriminants, Example. Future Software Project Management: Modern Project Profiles Next generation Software economics, modern Process transitions.
7. **Case Study**: The Command Center Processing and Display System- Replacement (CCPDS-R).

Text Books:

1. Software Project Management, Walker Royce, Pearson Education.
2. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, Tata McGrawHill.

Reference Books:

1. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly,2006
2. Head First PMP, Jennifer Greene & Andrew Stellman,O'Reilly,2007
3. Software Engineering Project Managent, Richard H.Thayer&Edward Yourdon, second edition, Wiley India,2004.

4. Agile Project Management, Jim Highsmith, Pearson education,2004
5. The art of Project management, Scott Berkun, O'Reilly,2005.
6. Software Project Management in Practice, Pankaj Jalote, Pearson Education,2002.

Outcomes:

- Describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project.
- Compare and differentiate organization structures and project structures.
- Implement a project to manage project schedule, expenses and resources with the application of suitable project management tools.

IT421	ELECTIVE-VI	E-COMMERCE	
Instruction: 3 Periods & 0 Tut/week Univ. Exam: 3 Hours			**Credits:3**
Internal: 30 Marks		**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Introduction:** Electronic Commerce-Frame Work, Anatomy of E-Commerce Applications, E- Commerce Consumer Applications, E-Commerce Organization Applications. Consumer Oriented Electronic Commerce - Mercantile Process Models.
2. **Electronic Payment Systems** – Types of Electronic Payment Systems, Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment Systems, Designing Electronic Payment Systems
3. **Electronic Data Inter Change**, Inter Organizational Commerce - EDI, EDI Implementation, Value Added Networks.
4. **Intra Organizational Commerce**, Macro Forces and Internal Commerce, Work Flow Automation and Coordination, Customization and Internal Commerce, Supply Chain Management.
5. **Business Cases for Document Library**, Digital Document Types, Corporate Data Ware-Houses.
6. **Advertising And Marketing**: Information Based Marketing, Advertising On Internet, Online Marketing Process, Market Research. Consumer Search and Resource Discovery, Information Search and Retrieval, Commerce Catalogues, Information Filtering.
7. **Multimedia-Key Multimedia Concepts**, Digital Video and Electronic Commerce, Desktop Video Processing, Desktop Video Conferencing.

Text Books:

1. Frontiers of Electronic Commerce, Kalakata and Whinston, Pearson.

Reference Books:

1. E-Commerce fundamentals and Applications, Hendry Chan, Raymond Lee, Tharam Dillon, Ellizabeth Chang, JohnWiley.
2. E-Commerce, S.Jaiswal,Galgotia.
3. E-Commerce, Efrain Turbon, Jae Lee, David King, H.MichaelChang.
4. E-Commerce - Business, Technology and Society, Kenneth C.Taudon, Carol GuyericoTraver.

CSE 422	**ENTREPRENEURSHIP** *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Basic Concepts of Management: Management:** Definition, Nature and Importance ; Functions of the Management; Levels of Management; F.W Taylor's Scientific Management; Henry Fayol's Principles of Management;
2. **Forms of Business Organizations:** Introduction, **Types of Business organizations: Private Sector**- Individual Ownership, Partnership, Joint stock companies and Co-Operative organizations**; Public sector**- Departmental Organizations, Public Corporations and Government Companies; The Joint sector Management.
3. **Production and operations Management:** Plant location- Factors to be considered in the selection of Plant location; Break - even analysis- Significance and managerial applications; Importance of Production Planning and Control and its Functions; Human Resource Management and Functions of Human Resource Manager (in brief); Functions of Marketing; Methods of Raising Finance.
4. **Entrepreneurship:** Definition, Characteristics and Skills, Types of Entrepreneurs, Entrepreneur vs. Professional Managers, , Growth of Entrepreneurs, Nature and Importance of Entrepreneurs, Women Entrepreneurs, Problems of Entrepreneurship.
5. **Entrepreneurial Development and Project Management:** Institutions in aid of Entrepreneurship Development, Idea generation: Sources and Techniques;, Stages in Project formulation ; Steps for starting a small enterprise - Incentives for Small Scale Industries by Government.

Text Books:

1. Sharma,S.C, and Banga, T.R., **Industrial Organization & Engineering Economics**, Khanna Publishers, Delhi, 2000.
2. Vasant Desai **The Dynamics of Entrepreneurial Development and Management (Planning for future Sustainable growth),**HImalayan Publishing House, 2018.

Reference Books:

1. Aryasri , A.R., **Management Science,** McGraw HIll Education (India Private Limited , New Delhi 2014.
2. Sheela, P. , and JagadeswaraRao, K., **Entrepreneurship**, Shree Publishing House, Guntur, Andhra Pradesh, 2017.

CSE 423	1G-4G MOBILE COMMUNICATION NETWORKS *Common with 6years integrated B.Tech(CSE)+M.Tech and B.Tech(IT)*	
Instruction: 3Periods/week, Univ. Exam: 3 Hours		**Credits: 3**
Internal: 30 Marks	**University Exam: 70 Marks**	**Total: 100 Marks**

Syllabus:

1. **Overview of Wireless Networks**, Introduction of Network Architecture and Design Issues and Key Trends in Wireless Networking. Three Generations of Cellular Networks, Trends in Wireless Technologies. Propagation Mechanisms, Propagation effects with mobile radio, Channel Classification.
2. **Generations of wireless mobile systems**, Cellular geometry, Introduction to cellular concept, Principle of operation of a cellular mobile system, Call transfer operation from one mobile phone to another, multiple access schemes, Analogue and digital cellular mobile systems.
3. **Cellular geometry**, Frequency reuse, Improving coverage and capacity in cellular systems, Cell splitting, Sectoring, Range extension by the use of repeaters, Microcell zone concept, Picocell zone concept.
4. **Structure of a wireless communication link**, Modulation and demodulation – Binary Phase shift Keying, Quadrature Phase Shift Keying, Quadrature Amplitude Modulation- (8 QAM & 16 QAM) and Binary Frequency Shift Keying.
5. **First generation** (1G), Second generation (2G), TDMA-based 2G standards, IS-95 (Code division multiple access (CDMA) or CDMA One standard), Two point five generation (2.5G), Third generation (3G) development, 3G Air interface technologies, 3G spectrum, Internet speeds of 2G, 2.5G, and 3G technologies, Limitations of 3G.
6. **Introduction to Wireless Systems & Standards**, CDMA, WCDMA evolution, OFDM, Wireless LANs.
7. **Introduction to 4G Networks**, Evolution of 4G, Objectives of 4G, Advantages of 4G network technology over 3G, Applications of 4G, 4G technologies, 4G software, Limitations of 4G.

Text Books:

1. G S RAO, Mobile Cellular Communication, Pearson Education, New Delhi, 2013
2. Rappaport T.S., "Wireless Communications; Principles and Practice", Prentice Hall, NJ, 2000.
3. Lee W.C.Y., "Wireless & Cellular Telecommunications", McGraw Hill, New York, 3e, 2005.

Reference Books:

1. Andrea Goldsmith, “Wireless Communications”, Cambridge University Press, 2005.
2. Simon Haykin& Michael Moher, “Modern Wireless Communications”, Pearson Education, 2007.
3. Andreas.F. Molisch, “Wireless Communications”, John Wiley – India, 2006.

GUIDELINES FOR DOING THE PROJECT WORK

1. Candidates can do their thesis work within the department or in any industry/research organization for one semester in the 4th year of their study. In case of project done in an industry/research organization, one advisor (Guide) should be from the department and one advisor (Co-Guide) should be from the industry/research organization.
2. Students should work in teams of 3 to 4 members and submit thesis on the project work done by them.

Format for Preparation of Project Thesis For B. Tech (IT):

Arrangement of Contents:
The sequence in which the project report material should be arranged and bound should be as follows:

1. Cover Page & Title Page
2. Bonafide Certificate
3. Abstract
4. Table of Contents
5. List of Tables
6. List of Figures
7. List of Symbols, Abbreviations and Nomenclature
8. Chapters
9. Appendices
10. References

*The table and figures shall be introduced in the appropriate places.