Course Curriculum

(Course Structure and Syllabi)

for

Bachelor of Technology

in

Computer Science and Engineering

(Second Year Onwards)

Department of Computer Science and Engineering

National Institute of Technology Hamirpur

Hamirpur (H.P) – 177 005 (India)

Curriculum for B Tech Programme

Course No.	Semester 3	Credits	Course Type
BS/Engg	Basic Sciences	3	Discipline core
	Engineering Courses	14	Discipline core
	Engineering Course (Lab)	2	Discipline core
Discipline Workshop	Basic Engineering Skills	1	Discipline core
Total		**20**	

Course No.	Semester 4	Credits	Course Type
	Engineering Course	13	Discipline core
	Engineering Course	3	Discipline Elective
	Engineering Course (Lab)	3	Discipline core
	LA/CA	1	Institute Elective
	Total	**20**	

Curriculum for B Tech Programme

Second Year													
3rd Semester							4th Semester						
SN	Code	Subject	L	T	P	C	SN	Code	Subject	L	T	P	C
1	MA-219	Probability and Applied Statistics	3	0	0	3	1	CS-221	Microprocessor & Interfacing	3	0	0	3
2	CS-212	Discrete Structures	3	0	0	3	2	CS-222	Operating Systems	3	0	0	3
3	CS-213	Object Oriented Programming	3	0	0	3	3	CS-223	Computer Organization and Architecture	3	0	0	3
4	CS-214	Data Structures	3	0	0	3	4	CS-224	Theory of Computation	3	1	0	4
5	CS-215	Computer Graphics	3	0	0	3	5	CS-241/CS-242/ CS-243	Discipline Elective-I	3	0	0	3
6	EC-219	Digital Electronics	2	0	0	2	6	CS-225	Microprocessor & Interfacing Lab	0	0	2	1
7	CS-217	Object Oriented Programming Lab	0	0	2	1	7	CS-226	Operating Systems Lab	0	0	2	1
8	CS-218	Data Structures Lab	0	0	2	1	8	CS-227	Computer Organization & Architecture Lab	0	0	2	1
9	CS-219	Computational Tools and Workshop	0	0	2	1	9	SA-201	LA/CA (NSS/NCC/Prayas etc)	0	0	2	1
		Total =				20			Total =				20

Discipline Elective (I)

1. CS-241: Data Communication
2. CS-242: Optimization Techniques
3. CS-243: Simulation & Modelling

Curriculum for B Tech Programme

Course No.	Semester 5	Credits	Course Type
	Open Elective	3	Institute Electives
	Engineering Course	**12**	Discipline core
	Engineering Course	3	Discipline Elective
	Engineering Course (Lab)	2	Discipline core
	HSS Course (**Non Circuital branches)**	**0**	Institute Core
Total		**20**	
Course No.	**Semester 6**	**Credits**	**Course Type**
	Engineering Course	**10/8**	Discipline Core
	Engineering Course	6	Discipline Elective
	Engineering Course	2	Stream Core
	Engineering Course (Lab)	2	Discipline Core
	HSS Course **(Circuital branches)**	**0/2**	Institute Core
Total		**20**	

Curriculum for B Tech Programme

Third Year													
5thSemester							6thSemester						
SN	Code	Subject	L	T	P	C	SN	Code	Subject	L	T	P	C
1	CS-301/302/303	Open Elective	3	0	0	3	1	CS-321	Digital Image Processing	3	0	0	3
2	CS-311	Analysis & Design of Algorithm	3	0	0	3	2	CS-322	Database Management Systems	3	0	0	3
3	CS-312	Compiler Design	3	0	0	3	3	CS-323	Software Engineering	2	0	0	2
4	CS-313	Computer Networks	3	0	0	3	4	CS-341/342/343	Discipline Elective-III	3	0	0	3
5	CS-314	Artificial Intelligence	3	0	0	3	5	CS-361/362/363	Discipline Elective-IV	3	0	0	3
6	CS-351/352/353	Discipline Elective-II	3	0	0	3	6	CS-381/382	Stream Core-I	2	0	0	2
7	CS-315	Compiler Design Lab	0	0	2	1	7	HS-321	Engineering Economics and Accountancy	2	0	0	2
8	CS-316	Computer Networks Lab	0	0	2	1	8	CS-324	Digital Image Processing Lab	0	0	2	1
							9	CS-325	Database Management Systems Lab	0	0	2	1
		Total =				20			Total =				20

Curriculum for B Tech Programme

Discipline Elective (II)

1. CS-351: Advance Operating System
2. CS-352: Graph Theory
3. CS-353: Information Retrieval

Discipline Elective (IV)

1. CS-361: Cloud Computing
2. CS-362: Statistical Computing
3. CS-363: Neural Network

Open Elective

1. CS-301: Data Structures
2. CS-302: Computer Networks
3. CS-303: Artificial Intelligence

Stream Core-I

1. CS-381: Distributed Systems
2. CS-382: Machine Learning

Discipline Elective (III)

1. CS-341: Game Theory
2. CS-342: Computer Vision
3. CS-343: Natural Language Processing

Curriculum for B Tech Programme

Course No.	Semester 7	Credits	Course Type
	Engineering Course	9	Discipline Core
	Engineering Course	3	Discipline Elective
	Engineering Course	4	Stream Core
	Engineering Course (Lab)	2	Discipline Core
Vocational Training	Engineering Course	2	Discipline Core
Total		**20**	

Course No.	Semester 8	Credits	Course Type
	UG Project*	12/12	Discipline elective
	Free Elective/Engineering Course/Open Elective Course (Courses available in other departments in the even semester)	**6**	Free Electives/Stream Elective (offered by Department/Institute Elective (Open Elective)
General Proficiency	Holistic Assessment	2	Institute Core
Total		**20**	

* (Students **opting for internship** will **complete the UG project** and the remaining credit requirements will be by fulfilled by opting **Free Elective Courses)**

Curriculum for B Tech Programme

Fourth Year													
7thSemester							8th Semester						
SN	Code	Subject	L	T	P	C	SN	Code	Subject	L	T	P	C
1	CS-411	Advance Computer Architecture	3	0	0	3	1	CS-461/462/ 463/464	*Free Elective / Stream Elective-I / Institute Elective	3	0	0	3
2	CS-412	Information Security & Privacy	3	0	0	3	2	CS-481/482/ 483/484	*Free Elective / Stream Elective-I / Institute Elective	3	0	0	3
3	CS-413	Data Warehousing & Data Mining	3	0	0	3	3	CS-498	General Proficiency				2
4	CS-431/432/433	Discipline Elective-V	3	0	0	3	4	CS-499	UG Project				12
5	CS-451/452	Stream Core-II	2	0	0	2							
6	CS-471/472	Stream Core-III	2	0	0	2							
7	CS-414	Information Security & Privacy Lab	0	0	2	1							
8	CS-415	Data Warehousing & Data Mining Lab	0	0	2	1							
9	CS-416	Summer Training	0	0	4	2							
		Total =				20			Total =				20

Curriculum for B Tech Programme

Discipline Elective (V)

1. CS-431: Information Theory and Coding
2. CS-432: Soft Computing
3. CS-433: Big Data Analytics

Stream Elective-I

1. CS-461: Reinforcement Learning
2. CS-462: Cyber Security
3. CS-463: Quantum Computing
4. CS-464: Networked Wireless System

Stream Core-II

1. CS-451: Advance Mobile Communication
2. CS-452: Deep Learning

Stream Elective-II

1. CS-481: Blockchain Technology
2. CS-482: Approximation Algorithms
3. CS-483: Parallel Computing
4. CS-484: Topics in Theoretical Computer Science

Stream Core-III

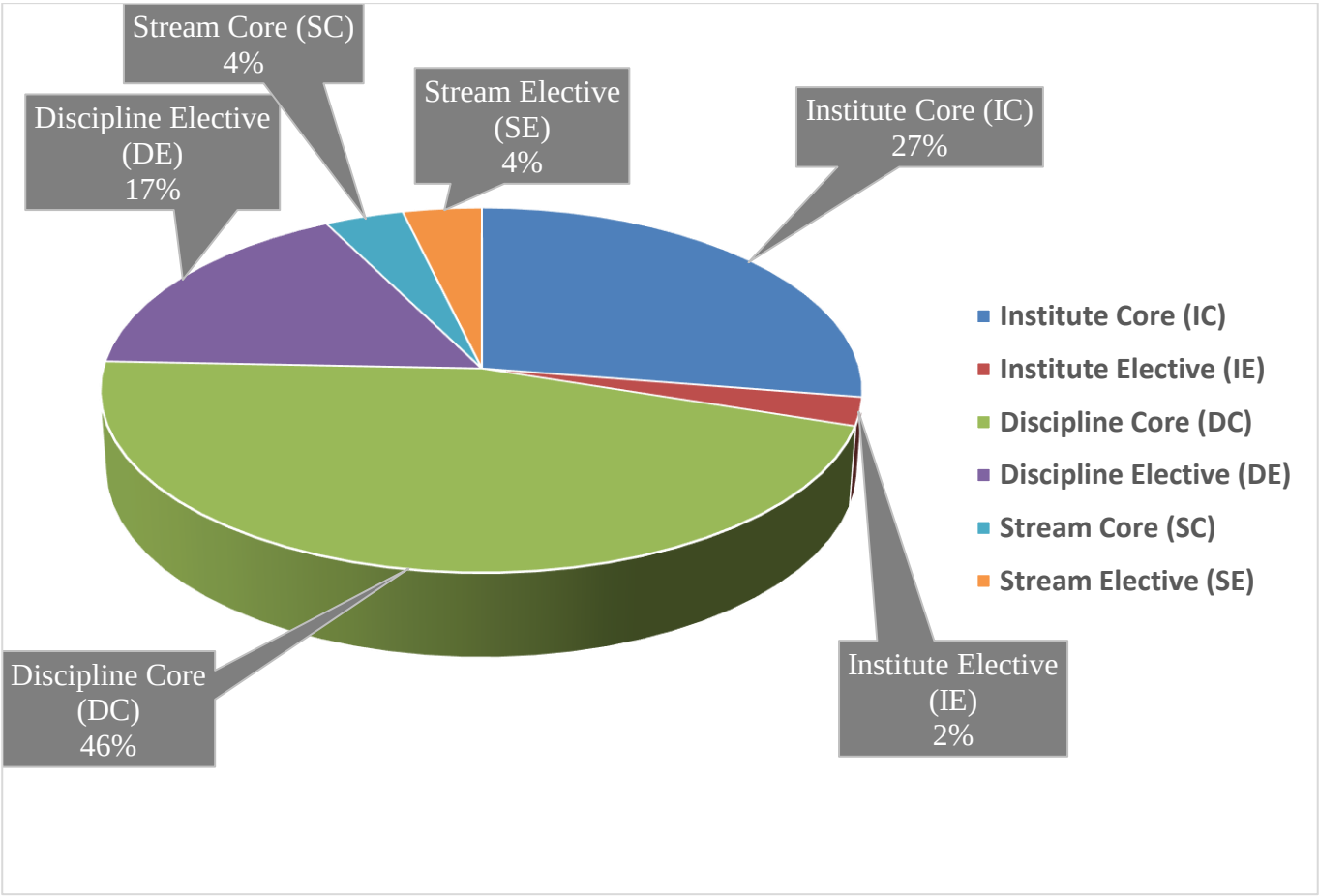
1. CS-471: Internet of Things
2. CS-472: Pattern Recognition

Types of Courses and credits in each Semester

	Semester								
Types of Courses	**1st**	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**	**Total**
IC	20	20	0	0	0	2	0	2	44
IE	0	0	0	1	3	0	0	6*	4-10*
DC	0	0	20	16	14	10	13	0	73
DE	0	0	0	3	3	6	3	12	27
SC	0	0	0	0	0	2	4	0	6
SE	0	0	0	0	0	0	0	6*	0-6*
Total	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**20**	**160**
Total									**160**

* Students are free to choose any combination out of Free Electives, IE and SE for 6 credits

Share of Credits based on the Requirement of the Programme



Course Name: **Probability and Applied Statistics**
Course Code: **MA-219**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- This course is designed to provide necessary basic concepts in probability, standard distributions and random processes which are widely applied in random signals, linear systems in communication engineering and IT fields.
- The syllabus also covers the concepts of Markovian and advanced queueing models which are essential to design and analyze computer networks.

Unit Number	Course Content	Contact Hours
UNIT-01	**Probability and Random Variables:** Axioms of probability, Conditional probability, Total probability, Baye's Theorem, Random Variables, Probability Mass Function, Probability Density Function-properties, Moments-Moment generating function and their properties.	6L
UNIT-02	**Standard Distributions:** Binomial, Poisson, Geometric, Negative Binomial, Uniform, Exponential, Gamma, Weibull and Normal distributions and their properties- Function of a random variable. Probability density function and its properties.	6L
UNIT-03	**Two Dimensional Random Variables:** Joint distributions, Marginal and conditional distribution, Covariance, Correlation and regression, Transformation of random variables, Central limit Theorem. Important properties of estimators, the method of maximum likelihood, the method of moments, and Bayesian estimation, Interval estimation for a single sample.	6L
UNIT-04	**Random Processes and Markov Chains:** Classification, Stationary process, Markov process, Poisson process, Birth and death process, Markov chains, Transition probabilities, Limiting distributions.	6L
UNIT-05	**Queueing Theory:** Markovian models, M /M/1, M/M/C, finite and infinite capacity, M/M/8 queues, Finite source model, M/G/1 queue (steady state solution only), Pollaczek-Khintchine formula, Special cases.	6L
UNIT-06	**Additional Topics:** The treatment of statistical methods with random sampling; data summary and description techniques, including stem-and-leaf plots, histograms, box plots, and probability plotting; several types of time series plots. Confidence intervals, for means, variances or standard deviations, proportions, prediction intervals, tolerance intervals. Hypothesis tests for a single sample.	7L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the fundamental concepts of probability and acquire knowledge of standard distributions which can describe real life phenomena.
CO2: Identify various distribution functions and acquire skills in handling situations involving more than one variable.
CO3: Analyze the various classifications of Random Processes and characterize phenomena which evolve with respect to time in a probabilistic manner.
CO4: Understand the basic characteristic features of a queuing system and acquire skills in analyzing queuing models.
CO5: Analyze a network of queues with Poisson external arrivals, exponential service requirements and independent routing.

Text Books:

1. Trivedi.K.S., "Probability and Statistics with Reliability, Queueing and Computer Science Applications", 2nd Edition, John Wiley and Sons, 2016.
2. Hwei Hsu, "Schaum˝s Outline of Theory and Problems of Probability, Random Variables and Random Processes", Tata McGraw Hill Edition, New Delhi, 2014.

References Books:

1. Robertazzi, "Computer Networks and Systems: Queueing Theory and Performance Evaluation", 3rd Edition, Springer, 2012.
2. Yates. R.D. and Goodman. D. J., "Probability and Stochastic Processes", 2nd Edition, Wiley India Pvt. Ltd., Bangalore, 2012.
3. Taha. H.A., "Operations Research", 8th Edition, Pearson Education, Asia, 2010.

Course Name: **Discrete Structures**
Course Code: **CS-212**
Course Type: **Discipline Core**

Contact Hours/Week: 3**L** Course Credits: **03**

Course Objectives

- To impart knowledge about disjunctive normal form and determine their validity by applying the rules and methods of propositional calculus Design algorithms and flowcharts.
- To introduce the fundamental concepts of common language to formal logic using the rules of propositional and predicate calculus and assess the validity of arguments.
- To enable the students to understand the logic recursion, basic structure and graph theory.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Preliminaries and Predicate Calculus:** Basic concepts of discrete mathematics and related problems**,** propositions and predicates, disjunction and conjunction, tautologies and contradiction, laws of equivalence, rules of substitution and transitivity, normal forms.	6L
UNIT-02	**Set Theory and Functions:** Basic concepts, Venn Diagrams, set operations, power set, methods of proof for sets, Relations and ordering, Types of relations, Graph and matrix of a relation, properties of a relation, Poset, Lattice.	4L
UNIT-03	Functions: definitions and notation, one to one, onto, one to one and onto, composition, identity and inverse, related results**.** Recurrence relations – partial and total recursion -problems.	4L
UNIT-04	**Graph Theory:** Basic concepts of graph theory, multigraphs and weighted graphs, walk path and circuits, Warshall"s algorithm: shortest path, Eulerian paths and circuits, Hamiltonian paths and circuits, factors of a graph and planar graphs, Graph colourings.	7L
UNIT-05	**Binary Trees:** Introduction, complete and extended binary tree, traversing binary tree, binary search tree, Heaps, Huffman's algorithm.	6L
UNIT-06	**Basics of Structures:** Mathematical induction, Algebraic structures properties, Semi group, Monoid, Group and Sub group - examples and standard results, generators and evaluation of powers, cosets and Langranges's theorem, rings, integral domains, fields.	9L

Course Outcomes

Upon successful completion of the course, the students will be able to:

CO1: Interpret statements presented in disjunctive normal form and determine their validity by applying the rules and methods of propositional calculus Design algorithms and flowcharts.

CO2: Rephrase statements from common language to formal logic using the rules of propositional and predicate calculus and assess the validity of arguments.

CO3: Understand and apply the properties of relations in computer science and engineering problems.

CO4: Understand graph problems and implement effective solutions.

Text Books:

1. Discrete Mathematical structures with applications to Computer Science by J. P. Tremblay and R Manohar, McGraw Hill.
2. Elements of Discrete Mathematics by C.L. Liu, McGraw Hill.
3. An Introduction to Discrete Mathematics by Roman, Steven, Saunders.

References Books:

1. Discrete Mathematics by Barnett, Steven, Addison Wesley.
2. Discrete Mathematics by Dossey, John A. et al, Addison-Wesley.
3. Discrete Mathematics and Its Applications by Kenneth H. Rosen, McGraw Hill.

Course Name: **Object Oriented Programming**
Course Code: **CS-213**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the concepts of Object-Oriented programing.
- To introduce the fundamental concepts relevant to Arrays, Pointers and Functions, Classes, Objects, etc.
- To enable the students to understand the standard library, exception handling, streams and files.

Unit Number	Course Content	Contact Hours
UNIT-01	**Concepts of Object-Oriented Programming:** Object Oriented Programming Paradigm, Basic concepts of OOP"s, Benefits of OOPS, and Introduction to object-oriented design and development, Design steps, Design example, Object oriented languages, Comparison of structured and object-oriented programming languages.	5L
UNIT-02	**Arrays, Pointers and Functions:** Arrays, Storage of arrays in memory, Initializing Arrays, Multi-Dimensional Arrays, Pointers, accessing array elements through pointers, passing pointers as function arguments, Arrays of pointers, Pointers to pointers, Functions, Arguments, Inline functions, Function Overloading Polymorphism**.**	6L
UNIT-03	**Classes and Objects:** Data types, operators, expressions, control structures, arrays, strings, Classes and objects, access specifiers, constructors, destructors, operator overloading, type conversion. **Storage classes:** Fixed vs Automatic declaration, Scope, Global variables, register specifier, Dynamic memory allocation. **Inheritance:** Inheritance, single Inheritance, Multiple Inheritance, Multi-level inheritance, hierarchical inheritance, hybrid inheritance, Virtual functions.	8L
UNIT-04	**Streams and Files:** Opening and closing a file, File pointers and their manipulations, Sequential Input and output operations, multi-file programs, Random Access, command line arguments.	5L
UNIT-05	**Templates:** Class templates, function templates, overloading of template functions. **Exception Handling and Graphics:** List of exceptions, catching exception, handling exceptions.	7L
UNIT-06	**Standard Template Library:** Standard Template Library, Overview of Standard Template Library, Containers, Algorithms, Iterators, Other STL Elements, Container Classes, General Theory of Operation, Vectors**.**	6L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the concepts of object-oriented paradigm and programming
CO2: Apply the concepts of polymorphism and inheritance
CO3: Implement exception handling and templates
CO4: AHandling of files and streams during programming

Text Books:

1. Object Oriented programming with C++ by E. Balagurusamy, Tata McGraw Hill.
2. The C++ programming Language by Bjarne Strustrup, Addison Wesley.
3. Object Oriented Analysis and Design with Applications by Grady Booch, Addison Wesley.

References Books:

1. The Complete Reference Visual C++ by Chris H. Pappas and William H. Murray, Tata McGraw Hill.
2. C++ Primer by S. B. Lippman, Josee Lajoie, Barbara E. Moo, Pearson Education.

Course Name: **Data Structures**
Course Code: **CS-214**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about linear and non-linear data structures as the foundational base for computer solutions to problems.
- To introduce the fundamental concepts relevant to binary trees, binary tree traversals, binary search trees and perform related analysis to solve problems
- To enable the students to understand various types of sorting algorithms.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Data types, data structures, abstract data types, the running time of a program, the running time and storage cost of algorithms, complexity, asymptotic complexity, big O notation, obtaining the complexity of an algorithm.	6L
UNIT-02	**Development of Algorithms:** Notations and Analysis, Storage structures for arrays - sparse matrices - structures and arrays of structures, Stacks and Queues: Representations, implementations and applications.	6L
UNIT-03	**Linked Lists:** Singly linked lists, Linked stacks and queues, operations on Polynomials, Doubly Linked Lists, Circularly Linked Lists, Operations on linked lists- Insertion, deletion and traversal, dynamic storage management – Garbage collection and compaction.	6L
UNIT-04	**Trees:** Basic terminology, General Trees, Binary Trees, Tree Traversing: in-order, pre-order and post-order traversal, building a binary search tree, Operations on Binary Trees - Expression Manipulations - Symbol Table construction, Height Balanced Trees (AVL), B-trees, B+-trees.	7L
UNIT-05	**Graphs:** Basic definitions, representations of directed and undirected graphs, the single-source shortest path problem, the all-pair shortest path problem, traversals of directed and undirected graphs, directed acyclic graphs, strong components, minimum cost spanning tress, articulation points and biconnected components, graph matching.	5L
UNIT-06	**Sorting and Searching Techniques:** Bubble sorting, Insertion sort, Selection sort, Shell sort, Merge sort, Heap and Heap sort, Quick sort, Radix sort and Bucket sort, Address calculation, Sequential searching, Binary Searching, Index searching, Hash table methods.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Interpret and compute asymptotic notations of an algorithm to analyze the time complexity
CO2: Use of linear and non-linear data structures as the foundational base for computer solutions to problems
CO3: Demonstrate the ability to implement various types of static and dynamic lists
CO4: Implement binary trees, binary tree traversals, binary search trees and perform related analysis to solve problems
CO5: Implement various types of sorting algorithms

Text Books:

1. An Introduction to Data Structures with applications by J.P. Tremblay and P.G. Sorenson, Tata McGraw Hill.
2. Data structures, Algorithms ad Applications in C++ by Sartaj Sahni, WCB/McGraw Hill.
3. Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft,, Addison Wesley.

References Books:

1. Data Structures using C by Y. Langsam, M. J. Augenstein and A. M. Tenenbaum, Pearson Education.
2. Data Structures – A Pseudocode Approach with C by Richard F. Gilberg and Behrouz A. Forouzan, Thomson Brooks /COLE.

Course Name: **Computer Graphics**
Course Code: **CS-215**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the computer graphics display technologies
- To introduce the fundamental concepts relevant to various 2D transformations, clipping algorithms to 2D primitives and Demonstrate the 3D transformation concepts to model an object.
- To enable the students to understand the implementation of various rendering algorithms.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Computer Graphics:** Overview of Graphics Systems, Display Devices, Hard copy Devices. Interactive Input Devices, Display Processors, The Graphical Kernel System, Output Primitives, Graphics hardware: Display technology, random scan, raster scan display processing, graphics software and standards.	5L
UNIT-02	**Raster Scan Graphics**: Points, lines, circles and ellipses as primitives, scan conversion algorithms for primitives, Fill area primitives including scan-line polygon filling, inside-outside test, boundary and flood-fill, character generation, line attributes, area-fill attributes, character attributers, antialiasing.	7L
UNIT-03	**Two-Dimensional Geometric Transformation & Viewing:** Basic Transformation, Translation, Rotation, Scaling, Other Transformation Reflection, Shear, Transformation functions, Window to viewport co-ordinate transformation, Clipping Operations, Point Clipping, Line Clipping, Polygon Clipping.	7L
UNIT-04	**Three- Dimensional Concepts & Object Representations:** Three Dimensional Display Methods, Parallel Projection, Perspective Projection, Translation, Rotation, Scaling, Composite Transformation, Three dimensional Transformation function, Polygon Surfaces, Curved Lines and surfaces, Bezier Curves and surfaces, B-Spline Curves and surfaces	8L
UNIT-05	**Visible Lines and Visible Surfaces:** Visual Realism, Hidden line and hidden surface removal: depth buffer algorithm, geometric computations, scan line coherence algorithms, area coherence algorithms, priority algorithm, visible surface detection concepts, back-face detection, depth buffer method, light sources, illumination methods (ambient, diffuse reflection, specular reflection), shading.	5L
UNIT-06	**Color models:** properties of light, XYZ, RGB, YIQ and CMY color models, modeling methods, Transparency, Refraction effects in transparent materials, Simple Transparency Models, Shadows, Texture.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the computer graphics display technologies.
CO2: Implement various 2D transformations.
CO3: Apply the clipping algorithms to 2D primitives.
CO4: Demonstrate the 3D transformation concepts to model an object.
CO5: Understand and implement various rendering algorithms.

Text Books:

1. Procedural Elements for Computer Graphics by D.F. Rogers, McGraw Hill.
2. Computer Graphics by Hearn and Baker, Prentice Hall of India.
3. Mathematical Elements for Computer Graphics by D.F. Rogers, McGraw Hill.

References Books:

1. Computer Graphics, with OpenGL by Hearn and Baker, Pearson Education.
2. Computer Graphics by Foley and van Dam, Person Education

Course Name: **Digital Electronics**
Course Code: **EC-219**
Course Type: **Discipline Core**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To impart knowledge about the concept of digital design, number system and codes
- To introduce the fundamental concepts related to design of combinational logic circuits
- To enable the students to understand the design of Sequential Circuits

Unit Number	Course Content	Contact Hours
UNIT-01	**Number System & Codes:** Binary, Octal, Hexadecimal number systems and their inter-conversion, Binary Arithmetic (Addition, Subtraction, Multiplication and Division), Diminished radix and radix compliments, BCD codes, 8421 code, Excess-3 code, Gray code, error detection and correction, Hamming code. **Logic Gates, Boolean Algebra & Logic Families:** Axiomatic definition of Boolean Algebra, Basic Theorems and properties of Boolean Algebra, Boolean Functions, Canonical and Standard forms, Digital Logic Gates. Various Logic Families like TTL and ECL etc., working and their characteristics, MOS and CMOS devices.	6L
UNIT-02	**Combinational Logic Design:** The map method, Two, Three, Four and Five variable maps, Sum of products and Product of Sums Simplification, NAND and NOR implementation, incompletely specified functions, Ex-OR functions, The tabulation method, Determination of Prime implicants, Selection of Essential Prime implicants, The cube notation, Sharp operation, Iterative Consensus, Generalized Consensus, Minimization of Multiple output switching functions, Determining Prime implicants using Generalized Consensus, Finding a Minimum cover, Breaking cyclic and similar structures.	6L
UNIT-03	**MSI and PLD Components:** Binary adder and subtractor, Multiplexers, Decoders / Demultiplexers, Read Only Memory, Programmable Logic Arrays, Programmable Array Logic. Implementation of Combinatorial Logic using these devices.	5L
UNIT-04	**Introduction to Sequential Logic:** Introduction, S-R Flip-flops, JK flip-flop, D flip-flop, T flip-flop, master slave flip-flop. Flip-flop excitation table, Classification of sequential circuits, Registers and A to D and D to A converter circuits, design & analysis of synchronous and asynchronous sequential circuits: Counters, Sequence Detector and Sequence Generator. **Semiconductor Memories:** Introduction, Memory organization, Classification and characteristics of memories, Sequential memories, ROMs, R/W memories. Content addressable memories, Programmable logic arrays, Charged-Coupled device memory.	7L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand about the concept of digital system
CO2: Apply principles of minimization techniques to simplify digital functions
CO3: Design and analyze the combinational electronic circuit based on digital logic
CO4: Design and analyze the sequential electronic circuit based on digital logic

Text Books:

1. M. Morris Mano, Digital Design, Prentice Hall of India.
2. Thomas Downs and Mark F Schulz, Logic Design with Pascal, Van Nostrand Reinhold.
3. Digital principle and applications Malvino and Leach- (TMH).

References Books:

1. Modern digital systems design Cheung (WPC).
2. Digital Electronics by JV Dutta, Tata McGrawHill.

Course Name:	**Object Oriented Programming Lab**
Course Code:	**CS-217**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To provide skills for designing object oriented programs.
- To provide skills for writing C++ programs.
- To enable the students to debug and implement programs involving different concepts of object oriented paradigm.

List of Experiments

1) To check that given number is even or odd using Bitwise Operator and to swap two numbers using third variable. To generate Fibonacci Series, to compute quotient and remainder and to print ASCII value of given character.
2) To calculate average of numbers using arrays and to display largest element of an array. To add two matrix using multi-dimensional arrays.
3) Searching in arrays, Array of Class, Operations on strings as arrays and to access array elements using pointers.
4) To swap numbers in cyclic order using call by reference and to check prime number by creating a function. To find the multiplication values and the cubic values using inline function. To calculate the area of circle, rectangle and triangle using function overloading.
5) To read and print students detail using class and object. To demonstrate example of friend function with class and to concatenate two strings using operators. To Increment ++ and decrement - - variables by using operator overloading and to subtract complex number using binary operator.
6) Write a program to demonstrate the single inheritance, multilevel inheritance, multiple inheritances, hybrid inheritance and hierarchical inheritance. Write a program that show that two files named 'Source 1' and 'Source 2' contains sorted list of integers. Read the content of both the files and stores the merged list in sorted form in a new file named 'Target'.
7) Write a function template to perform linear search in an array. Write a program containing a possible exception. Use a try block to throw it and a catch block to handle it properly.
8) Write a program using the find() algorithm to locate the position of a specified value in a sequence container.
9) Write a program using the algorithm count() to count how many elements in a container have a specified value.
10) Write a program to implement Vector in STL.

Note: *The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.*

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Identify and abstract the programming task involved for a given problem.
CO2: Design and develop object-oriented programming skills.
CO3: Trace and debug a program.

Course Name:	**Data Structures Lab**
Course Code:	**CS-218**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To provide skills for designing & writing algorithms.
- To provide skills for writing C/C++ programs.
- To enable the students to debug programs.

List of Experiments

1. Given an array X. Compute the array A such that A[i] is the average of elements X[0]. . . X[i], for i = 0....n − 1. You can solve this with two methods, one with $O(n^2)$ and one with O(n) time complexities. Compare time complexities of both the methods by experimental approach plotting graph between 'n' and time taken for execution.
2. Write a program to sort an array (make a dynamic array) using Bubble sort. Use 1-bit variable FLAG to signal when no interchange take place during pass. If FLAG is 0 after any pass, then list is already sorted and there is no need to continue.
3. WAP to search an ITEM (integer) in an array using binary search, if FOUND then delete that item from array and if NOT FOUND than insert that item in position such that array remain sorted.
4. Implement linked list and insert and delete an element into the list.
5. Evaluate a postfix algebraic expression with the help of stack.
6. Implement a queue using arrays and linked list.
7. Implement a binary tree and implement any traversal technique as you like.
8. Implement a binary Search Tree and insert and delete a node in the BST.
9. Implement Max Priority queue using Max Heap.
10. Implement a graph and find transpose of a graph where Transpose of a directed graph G is another directed graph on the same set of vertices with all of the edges reversed compared to the orientation of the corresponding edges in G. That is, if G contains an edge (u, v) then the converse/transpose/reverse of G contains an edge (v, u) and vice versa. Implement it with the help of adjacency list and adjacency matrix.
11. Implement Quick Sort, Merge Sort, Insertion Sort and Selection Sort.

Note: *The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.*

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: To gain knowledge of popular available data stuctures.
CO2: To develop programming skills in students.
CO3: To impart knowledge of syntax and semantics of basic languages.

Course Name:	**Computational Tools and Workshop**
Course Code:	**CS-219**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To provide skills for designing and development of web pages, websites and portals.
- To enable students to work with various web programming/scripting languages.
- To provide skills to work towards solution of real-life problems.

List of Experiments

1. Write a javascript program to implement the use of Variables and Assignments
2. Write a javascript program to implement user-defined functions
3. Write a javascript program to implement Comparisons and Control Flow
4. Write a javascript program to implement Built in Functions & Methods.
5. Write a javascript program to implement Object Oriented JavaScript.
6. Write a javascript program to implement Dynamic Web Page Creation and Javascript Graphics
7. Write a javascript program to implement javascript Loops, Events, IFRAME and jQuery.
8. Write a python program to show database connectivity with Python
9. Write a python program for using python with internet services such as sending email, reading news, etc.
10. Write a python program for web client programming such as parsing, scraping and building crawlers.
11. Write a python program to parse, write and search XML files and elements.
12. Web development with python Flask and Django

Note: *The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.*

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the basic concepts related to web application development.

CO2: Apply JavaScript and Python to develop web-based solutions.

CO3: Apply various database & related concepts for use in web application development.

Course Name: **Microprocessor & Interfacing**
Course Code: **CS-221**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about microprocessors and internal architecture of 8085/8086 microprocessors.
- To introduce the fundamental concepts relevant to assembly language programs to solve a given problem.
- To enable the students to understand and implement programs to interface the 8085 microprocessors with peripheral devices.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Microprocessor:** History and Evolution, types of microprocessors, 8085 Microprocessor, Architecture, Bus Organization, Registers, ALU, Control section, Instruction set of 8085, Instruction format, Addressing modes, Types of Instructions.	6L
UNIT-02	**Assembly Language Programming and Timing Diagram:** Assembly language programming in 8085, Macros, Labels and Directives, Microprocessor timings, Microinstructions, Instruction cycle, Machine cycles, T states, State transition diagrams, Timing diagram for different machine cycles.	6L
UNIT-03	**Serial I/O, Interrupts and Comparison of Contemporary Microprocessors:** Serial I/O using SID, SOD. Interrupts in 8085, RST instructions, Issues in implementing interrupts, Multiple interrupts and priorities, Daisy chaining, Interrupt handling in 8085, Enabling, disabling and masking of interrupts. Brief comparison of contemporary 8-bit microprocessors like Z-80, M68000 with 8085.	7L
UNIT-04	**Data Transfer techniques:** Data transfer techniques, Programmed data transfer, Parallel data transfer using 8155. Programmable parallel ports and handshake input/output, Asynchronous and Synchronous data transfer using 8251A. Programmable interrupt controller 8259A. DMA transfer, cycle stealing and burst mode of DMA, 8257 DMA controller.	7L
UNIT-05	**Microprocessor Interfacing Techniques:** Interfacing memory and I/O devices, addressing memory, interfacing static RAMs, Interfacing and refreshing dynamic RAMs, interfacing a keyboard, Interfacing LED and seven segment displays, interfacing a printer, Interfacing A/D converters, D/A converters.	5L
UNIT-06	**Architecture of 8086:** Memory Address space and data organization, - segment registers and memory segmentation, generating memory addresses, IO address space, addressing modes, Comparison of 8086 and 8088, minimum mode maximum mode, system timing, introduction to Pentium and further series of microprocessors.	4L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the concepts of microcomputer and microprocessors and internal architecture of 8085/8086 microprocessors
CO2: Write and implement assembly language programs to solve a given problem
CO3: Write and implement programs to interface the 8085 microprocessors with peripheral devices

Text Books:

1. Fundamentals of Microprocessors and Microcomputers by B. Ram, Dhanpat Rai and Sons.
2. Microprocessor Architecture, Programming and applications with the 8085/8080A by R.S. Gaonkar, Wiley.
3. Microprocessors& Interfacing by Douglas V Hall, McGraw Hill.

References Books:

1. Microprocessors and Digital Systems by Douglas V Hall, McGraw Hill.
2. Introduction to Microprocessor by A.P. Mathur, Tata McGraw Hill.

Course Name: **Operating Systems**
Course Code: **CS-222**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the concepts of operating system and its management.
- To introduce the fundamental concepts scheduling of processes for a given problem instance.
- To enable the students to understand memory management techniques and implement replacement algorithms and understand and implement file systems.

Unit Number	Course Content	Contact Hours
UNIT-01	**Evolution of Operating Systems:** Evolution of operating systems, Types of operating systems. The process concept, system programmer's view of processes, operating system's views of processes, operating system services for process management.	6L
UNIT-02	**CPU Scheduling:** Scheduling concepts, scheduling algorithms, algorithm evaluation, multiple processor scheduling, real time scheduling.	6L
UNIT-03	**Concurrent Programming and Deadlocks:** Critical regions, Conditional critical regions, Monitors, Inter-process communication, Messages, Pipes, Semaphores, Modularization, Synchronization, Concurrent languages. Deadlocks: Characterization, Prevention, Avoidance, Detection and Recovery, Combined approach to Deadlock Handling, precedence graphs.	6L
UNIT-04	**Memory Management:** Memory Management, Contiguous allocation, static-swapping, overlays, dynamic partitioned memory allocation, demand paging, page replacement, segmentation. Non-contiguous allocation, paging, Hardware support, Virtual Memory.	7L
UNIT-05	**File Systems:** A Simple file system, General model of a file system, Symbolic file system, Access control verification, Logical file system, Physical file system, Allocation strategy module, Device strategy module, I/O initiators, Device handlers, Disk scheduling.	6L
UNIT-06	**Networks, Security and Design Principles:** Network operating system, distributed operating system, external security, operational security, password protection, access control, security kernels, hardware security, layered approach, design principle.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand and analyze the concepts of operating system and its management.
CO2: Illustrate the scheduling of processes for a given problem instance.
CO3: Identify the dead lock situation and apply appropriate solution.
CO4: Analyze memory management techniques and implement replacement algorithms.
CO5: Understand and implement file systems.

Text Books:

1. Operating System Concepts by J.L. Peterson and A. Silberchatz, Addison Wesley.
2. An Introduction to Operating System by Harvey M. Dietel, Addison Wesley.
3. Operating Systems - A Design Oriented Approach by C. Crowley, Irwin Publishing.

References Books:

1. Operating systems by W. Stallings, Prentice Hall.
2. Modern Operating system by A.S. Tanenbaum, Prentice Hall of India.

Course Name: **Computer Organization and Architecture**
Course Code: **CS-223**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the organization of any computing system.
- To introduce the fundamental concepts relevant to design instruction set architectures and develop their micro architectures.
- To enable the students to understand the factors various caching and architecture memory system architectures and instruction level parallelism.

Unit Number	Course Content	Contact Hours
UNIT-01	**General System Architecture:** Stored Program control concept (Von-Newman architecture principle), Flynn's Classification of computers (SIMD, MISD, MIMD), Structure organization (CPU, Caches, Main memory, Secondary memory unit & I/O), Register Transfer Operation, Micro-operation	4L
UNIT-02	**General System Architecture:** Instruction Set Architecture (Instruction set based classification of processor i.e. RISC, CISC), Addressing Modes, Operation instruction set (Arithmetic & logical, Data transfer, Control flow), Instruction set format	3L
UNIT-03	**Processor Design:** Arithmetic and logic unit, Stack organization, CPU Architecture types, Accumulator Based- Register, Stack-Memory, Register, Detailed data path of a typical register based CPU, Fetch, Decode, and Execute Cycle. Addition and Subtraction, Multiplication Algorithms (Booth's Multiplication Algorithm), Division Algorithm, Floating point arithmetic operations.	8L
UNIT-04	**Control Design:** Microprogrammed and Hard-wired control options, Hard-wired design methods, State table method, Multiplier control, CPU control unit. Microprogrammed, Basic concepts, control Memory, Address Sequencing	7L
UNIT-05	**I/O Organization & Memory Hierarchy:** Programmed, Interrupt driven & Direct Memory Access, Synchronous & synchronous data transfer, The need for Memory Hierarchy, locality of reference principle, Memory Hierarchy, Cache, Cache mapping techniques: Associative, Set associative, direct mapping, Main and Secondary Memory, Memory parameters, access cycle time, cost per unit.	7L
UNIT-06	**Introduction to Parallelism:** Goals of parallelism, Instruction level parallelism, pipelining, Hazards, Data hazard, Structural hazard, Control hazard, super scaling, Processor level parallelism, Multiprocessor system overview.	7L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Appreciate macro organization of any computing system
CO2: Design instruction set architectures and develop their micro architectures
CO3: Understand various digital arithmetic algorithms
CO4: Analyze various caching and architecture memory system architectures
CO5: Understand instruction level parallelism

Text Books:

1. Computer Organization and Architecture, Designing for Performance by William Stallings, Pearson Education.
2. Computer System Architecture by M. Morris Mano, Prentice Hall of India.
3. Computer Architecture: A quantitative Approach by David A. Patterson and John. L. Hennessy, Morgan Kaufmann.

References Books:

1. Structured Computer Organization by Andrew S. Tanenbaum and Todd Austin, Prentice Hall of India.
2. Advanced Computer Architecture: Parallelism, Scalability, Programmability by Kai Hwang, Tata McGraw Hill.

Course Name: **Theory of Computation**
Course Code: **CS-224**
Course Type: **Discipline Core**

Contact Hours/Week: **3L+1T** Course Credits: **04**

Course Objectives

- To impart knowledge about the basic concept of finite automata and formal languages.
- To introduce the fundamental concepts relevant to context free grammars and ability to construct grammars for specific languages.
- To enable the students to understand the Turing Machine and push down automata.
- To enable the students the concept of decidable and undecidable problems and computational complexity.

Unit Number	Course Content	Contact Hours
UNIT-01	**Machines:** Basic machine, FSM, Transition graph, Transition matrix, Deterministic and non-deterministic FSMS, Equivalence of DFA and NDFA, Mealy and Moore machines, minimization of finite automata, Two-way finite automata.	6L
UNIT-02	**Regular Sets and Regular Grammars**: Alphabet, words, Operations, Regular sets, Finite automata and regular expression, Pumping lemma and regular sets, Application of pumping lemma, closure properties of regular sets.	4L
UNIT-03	**Formal Grammars and Languages:** Basic definitions and examples of languages, Chomsky hierarchy, Regular grammars, context free & context sensitive grammars, context free languages, non-context free languages.	5L
UNIT-04	**Chomsky Normal Forms**, binary operations on languages. Simplification of CFG, Elimination of Useless symbols, Unit productions, Null productions, Greiback Normal form, Chomsky Normal form – Problems related to CNF and GNF.	6L
UNIT-05	**Pushdown Automata and Turing Machines:** Basics of PDA, Acceptance by PDA, Construction of PDA from CFG.	4L
UNIT-06	**TM model**, representation and languages acceptability of TM Design of TM, Variations of the Turning Machine, Universal TM and Other modification, Composite and iterated TM, Post Machine, Church Thesis	5L
UNIT-07	**Computability and Undecidability:** TM Languages: Turing Acceptable, Turing Decidable, Unrestricted Grammar, Modified Chomsky Hierarchy Basic concepts, properties of Recursive and Recursive Enumerable Languages, Undecidability, Reducibility, Post Correspondence Problem.	8L
UNIT-08	**Computational Complexity Theory:** Definition, linear speed-up, tape compression & reduction in number of tapes, Hierarchy Theorem, Relation among complexity measures, Transition lemmas and non-deterministic hierarchies, properties of general complexity measures, the gap, speed-up, union theorem, Automatic complexity theorem.	8L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the basic concept of finite automata and formal languages
CO2: Demonstrate understating of regular expressions and their connection to FSM
CO3: Demonstrate understating of context free grammars and ability to construct grammars for specific tasks
CO4: Demonstrate understanding of Turing Machine and push down automata
CO5: Understand the concept of undecidable problems and computational complexity.

Text Books:

1. Introduction to Automata theory, Languages and Computation by John E. Hopcroft, Jeffery Ullman, Narosa Publishers.

2. Introductory Theory of Computer Science by E.V. Krishnamurthy, East West Press.

3. Theory of computer Science by K.L.P. Mishra, Prentice Hall of India.

Reference Books:

1. An Introduction to Formal Languages and Automata by Peter Linz, Jones & Bartlett Learning.

2. Introduction to Languages and the Theory of Computation by John C. Martin, Tata McGraw Hill

Course Name: **Data Communication**
Course Code: **CS-241**
Course Type: **Discipline Elective -I**

Contact Hours/Week**: 3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the principles of data communication.
- To introduce the fundamental concepts of signal and transmission, multiplexing, etc.
- To enable the students to understand about transmission media and errors.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Data Communication, communication model, Networks, Protocols and Standards, Introduction to OSI and TCP/IP protocol architectures.	**6L**
UNIT-02	**Signal and transmission:** Analog and Digital data transmission, Analog and digital signals, periodic and non-periodic signals, **Periodic Analog Signals:** Sine wave, phase, wavelength, Time and Frequency Domains, Composite Signals, **Digital Signals:** Bit Rate, Bit length, transmission.	**7L**
UNIT-03	**Modulation and multiplexing techniques:** Parallel and serial transmission, Amplitude modulation (AM), Frequency modulation (FM), Phase modulation (PM)	**6L**
UNIT-04	**Digital modulation techniques** (ASK, FSK, PSK, QAM), pulse code modulation (PCM), delta modulation (DM), and adaptive delta modulation (ADM). Demodulation, Frequency division multiplexing (FDM), Time division multiplexing (TDM), Code division multiplexing (CDM).	**5L**
UNIT-05	**Transmission media**: Guided Media, Unguided Media (wireless), transmission impairments, noiseless and noisy channel, channel capacity, bandwidth, throughput, latency, jitter, QoS, Media Comparison, PSTN, Switching	**6L**
UNIT-06	**Error detection and correction**: Types of Errors, Detection, Parity Check, Vertical Redundancy Check Longitudinal Redundancy Check, Cyclic Redundancy Check, Checksum, Error Correction	**5L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO 1: Understand the basics of data communications.

CO 2: Understand data transmission techniques and use them in modulation and multiplexing.

CO 3: Describe different transmission media and their protocols.

Books and References

1. Data communication & Networking by Bahrouz Forouzan. TMH
2. Data & Computer Communications by William Stallings, PHI
3. Computer Networks by Andrew S. Tanenbaum
4. Communication system by Simon Hykin, John Wiley

Reference Book:

1. Introduction to Data Communication & Networking, Wyne Tomasi, Pearson.
2. Computer Networking, James Kurose, Pearson.

Course Name: **Optimization Techniques**
Course Code: **CS-242**
Course Type: **Discipline Elective -I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- This course intends to present a thorough treatment of optimization techniques with specific emphasis on modern applications.
- This will provide students with a sound background in the area and benefit those who wish to pursue doctoral or master level theses in this subject, or apply these techniques to their own areas.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Optimization:** Historical Development, Engineering applications of Optimization, Design vector and constraints, Constraint surface, Objective function, Classification of Optimization Problems.	4 L
UNIT-02	**Classical Optimization Techniques:** Single variable optimization, Constrained and unconstrained multi-variable optimization, Direct substitution method, Lagrange's method of multipliers, Karush-Kuhn-Tucker conditions.	6 L
UNIT-03	**Linear Programming:** Statement of an LP problem, Graphical Solution of an LP problem, Simplex method, Dual simplex method. **Non-linear Programming:** One-dimensional minimization method Unimodal function, Unrestricted search, Exhaustive search, Dichotomous search, Interval halving method, Fibonacci method, Golden section method, Direct root methods.	9 L
UNIT-04	**Non-linear Programming: Unconstrained Optimization Techniques** Direct Search Methods: Random search methods, Grid search method, Univariate method, Hookes and Jeeves' method, Powell's method Indirect Search Methods: Steepest descent method, Fletcher-Reeves method, Newton's method.	7 L
UNIT-05	**Non-linear Programming: Constrained Optimization Techniques** Direct Methods: Random search method, Sequential linear programming Indirect methods: Transformation techniques, Exterior penalty function method, Interior penalty function method.	7 L
UNIT-06	**Evolutionary Algorithms** An overview of evolutionary algorithms, Simulated annealing algorithm, Genetic algorithm, Particle swarm optimization	4 L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Identify optimization playing a critical role even in contemporary areas such as decision and control, signal processing, and machine learning

CO2: Understand basic theoretical principles for formulation of optimization models and its solution.

CO3: Learn the unified and exact mathematical basis as well as the general principles of various soft computing techniques.

CO4: Apply detailed theoretical and practical aspects of intelligent modelling, optimization and control of linear and non-linear systems.

Text Books

1. Optimization for Machine Learning, S. Sra, S. Nowozin, and S. Wright (eds). The MIT Press,
2. Statistical Learning with Sparsity: the Lasso and Generalizations, T. Hastie, R. Tibshirani and M. J. Wainwright, Chapman and Hall/CRC Press.
3. Introduction to Online Convex Optimization. E. Hazan.

References Books

1. Introductory lectures on convex optimization, Y. Nesterov, Kluwer-Academic, 2003.
2. Convex Optimization, S. Boyd and L. Vandenberghe, The Cambridge University Press.
3. Nonlinear programming, D. Bertsekas, Athena Scientific.

Course Name: **Simulation and Modelling**
Course Code: **CS-243**
Course Type: **Discipline Elective-I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To introduce the students to the fundamental concepts and principles of simulation modeling.
- To enable the students to build credible simulation models using various approaches and methods.
- To impart knowledge in respect of analysis of simulation output data to evaluate performance criteria.

Unit Number	Course Content	Contact Hours
UNIT-01	**Fundamentals:** Definition and reasons for simulation, Continuous (time-oriented) and discrete (event) systems, Modeling/programming simple deterministic systems, Rates and system dynamics.	**6L**
UNIT-02	**Concepts in Simulation:** Stochastic variables, discrete vs continuous probability, Monte Carlo Simulations; Monte Carlo methods, Normally distributed random numbers, Monte Carlo V/S Stochastic Simulations.	**6L**
UNIT-03	**Queuing Models:** Single server queuing system, introduction to arrival and departure time, flowcharts for arrival and departure routine. Event graphs of queuing model. Determining the events and variables, Event graphs for inventory model.	**6L**
UNIT-04	**Random Numbers:** Introduction to Random Numbers, Importance of Random Numbers in Simulation, MidSquare random number generator, Residue method, Arithmetic Congruential generator, Testing Numbers for Randomness, Chi-Square Test.	**8L**
UNIT-05	**Discrete Event System Simulation:** Discrete events; representation of time; queues and servers; generation of arrival patterns; resource seizing; departures simulation of a telephone system and computer networks.	**8L**
UNIT-06	**Simulating components of an operating system:** delayed calls; modeling policies; priority queues; tasks; gathering statistics; counters and summary statistics; measuring utilization and occupancy; recording distributions and transit times. Introduction to a Simulation Languages Simulation in C++, GPSS/ MATLAB/Network Simulators.	**8L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the role of simulation in problem-solving and decision-making.
CO2: Develop skills in building valid and credible simulation models.
CO3: Apply simulation techniques to solve real-world problems in various domains.

Text Books:

1. Discrete Event System Simulation by Jerry Banks and John Carson, PHI.
2. System Simulation by Geoffrey Gordon, PHI.
3. System Modeling and Simulation by Frank L. Severance, Wiley.

Reference Books:

1. Simulation Modeling and Analysis by Averill M. Law and W. David Kelton, McGraw Hill.
2. Handbook of Simulation: Principles by Jerry Banks, Wiley.
3. Introduction to Probability Models by Sheldon M. Ross, Academic Press.

Course Name:	**Microprocessor & Interfacing Lab**
Course Code:	**CS-225**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To impart knowledge about microprocessors and internal architecture of 8085/8086 microprocessors.
- To introduce the fundamental concepts relevant to assembly language programs to solve a given problem.
- To enable the students to understand and implement programs to interface the 8085 microprocessor with peripheral devices.

List of Experiments

1. Write a program to transfer an 8 bit data from register B to accumulator. Write a program to transfer an 8 bit data from memory location to accumulator.
2. Write a program to transfer the data byte to accumulator by using 1) MOV, 2) LDA, 3) LDAX.
3. Write a program to find out the 2's complement of a given 8 bit number. Write a program to find out the 2's complement of a given 16 bit number. Write a program to find out the numbers of odd and even numbers from an array of numbers. Write a program to add two 8 bit numbers considering the carry.
4. Write a program to subtract two 16 bit numbers considering the borrow. Write a program to mask the lower nibble of an 8 bit numbers.
5. Write a program to shift an 8 bit number left by 2 bit. Write a program to shift an 16 bit number by 2 bit without using rotate instruction. Write a program to find out the square of an 8 bit using look up table. Find out the smallest and largest from a array of numbers.
6. Find out the sum of a series of 8 bit numbers. Find out the smallest and largest from a array of numbers. Find out the sum of a series of 8 bit numbers. Sort array of numbers in ascending and descending order.
7. Write a program to multiply two 8 bit numbers. Transfer block of N-bytes from source to destination in forward and reverse order.
8. Unpack a BCD number. Pack two BCD numbers. Write a program to convert two BCD numbers in memory location to the equivalent HEX number.
9. Write a program to find out the factorial of a given number. Write a program to interface 8255A with 8085 or 8086 microprocessor kit.
10. Study of 8251 microcontroller kit and Interfacing of keyboard with 8251 microcontroller

Note: *The concerned Course Coordinator will prepare the actual list of experiments/problmes at the start of semester based on above generic list.*

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the concepts of microcomputer and microprocessors and internal architecture of 8085/8086 microprocessors.
CO2: Write and implement assembly language programs to solve a given problem.
CO3: Write and implement programs to interface the 8085 microprocessor with peripheral devices.

Course Name:	**Operating System Lab**
Course Code:	**CS-226**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To enlighten the student with knowledge about fundamental operating system abstractions.
- To impart knowledge about operating system resource management techniques.
- To enable the students to perform OS tasks in a Linux environment.
- To introduce the students to the different configurations in respect of different operating systems.

List of Experiments

1. Practice various commands/utilities in MS Windows and Linux OS for the following:
    - File management: create, copy, delete, link, display content, printing
    - Directory handling: create, delete, relocate, display content, sharing
    - Security features: setting/resetting file permissions to read, write, execute
    - Network configuration
2. Learn the top command to display resource utilization statistics of processes.
3. Simulate various CPU scheduling algorithms (FCFS, SJF-preemptive & non-preemptive, Priority Scheduling-preemptive & non-preemptive and Round Robin). Run your simulator for each scheduling policy, with a variety of quantum values. For each version of the simulator, for each input data file, for each quantum value, plot the completion time, throughput, average job elapsed time and average job waiting time. Analyze the behavior of your scheduler.
4. Write a program to simulate Banker's algorithm for deadlock handling.
5. Simulate memory allocation algorithms (First Fit and Best Fit). Simulate different page replacement strategies (FIFO, LRU and Optimal) and compare their performance.
6. Using the fork(…) system call, write a program in which the parent creates a child and the child creates a grandchild. Synchronize the processes so that the grandchild first displays a message (and then terminates), the child displays a message (and then terminates), and finally the parent displays a message.
7. Simulate (at least three) disk scheduling protocols. Write a program to display the IP address and MAC address of a machine.
8. Write script to implement background process that will continually print current time in upper right corner of the screen, while user can do his/her normal job at prompt.
9. Write shell script to show various system configuration like
    - Currently logged user and his log name, your current shell, your home directory, your current path setting, your current working directory, Show Currently logged number of users, About your OS and version, release number, kernel version
    - Show all available shells, Show computer CPU information like processor type, speed, etc.
    - Show memory information, Show hard disk information like size of hard-disk, cache memory, model, etc.
    - File system (Mounted)
10. Installation and configuration of MS Windows and Linux OS.

Note: The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Implement elementary UNIX system commands.
CO2: Devise programs to test synchronization problems.
CO3: Design and develop user level thread library.
CO4: Design and implement file system.

Course Name:	**Computer Organization and Architecture Lab**
Course Code:	**CS-227**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- Understanding the behavior of logic gates adders, decoders, multiplexers and flipflops.
- Should be able to understand concept of cache memory.

List of Experiments

1. Write a program to implement Ripple Carry Adder. Write a program to implement Carry-Look-Ahead Adder
2. Write a program to implement Wallace Tree Adder. Write a program to implement Synthesis of Flip Flops
3. Write a program to implement Registers and Counters
4. Write a program to implement Combinational Multipliers
5. Write a program to find the factorial of a number in MIPS, C & Python and check the maximum data size for which factorial could be calculated. Find out how the overflow conditions are handled in each architecture.
6. Write a program to implement Arithmetic Logic Unit
7. Write a program to understand register and memory allocation in register in MIPS architecture.
8. Write a program to implement Karnaugh Map
9. Write a program to develop FSM for CPU.
10. Implement the cache memory approaches using ParaCache simulator

Note: *The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.*

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Identify, compare and assess issues related to ISA, memory, control and I/O functions.
CO2: Apply various arithmetic operations and processor design methods.
CO3: Analyze the architecture and functionality of central processing unit.

Course Name: **Data Structures**
Course Code: **CS-301**
Course Type: **Institute Elective-I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about linear and non-linear data structures as the foundational base for computer solutions to problems.
- To introduce the fundamental concepts relevant to binary trees, binary tree traversals, binary search trees and perform related analysis to solve problems
- To enable the students to understand various types of sorting algorithms.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Data types, data structures, abstract data types, the running time of a program, the running time and storage cost of algorithms, complexity, asymptotic complexity, big O notation, obtaining the complexity of an algorithm.	6L
UNIT-02	**Development of Algorithms:** Notations and Analysis, Storage structures for arrays - sparse matrices - structures and arrays of structures, Stacks and Queues: Representations, implementations and applications.	6L
UNIT-03	**Linked Lists:** Singly linked lists, Linked stacks and queues, operations on Polynomials, Doubly Linked Lists, Circularly Linked Lists, Operations on linked lists- Insertion, deletion and traversal, dynamic storage management – Garbage collection and compaction.	6L
UNIT-04	**Trees:** Basic terminology, General Trees, Binary Trees, Tree Traversing: in-order, pre-order and post-order traversal, building a binary search tree, Operations on Binary Trees - Expression Manipulations - Symbol Table construction, Height Balanced Trees (AVL), B-trees, B+-trees.	7L
UNIT-05	**Graphs:** Basic definitions, representations of directed and undirected graphs, the single-source shortest path problem, the all-pair shortest path problem, traversals of directed and undirected graphs, directed acyclic graphs, strong components, minimum cost spanning tress, articulation points and biconnected components, graph matching.	5L
UNIT-06	**Sorting and Searching Techniques:** Bubble sorting, Insertion sort, Selection sort, Shell sort, Merge sort, Heap and Heap sort, Quick sort, Radix sort and Bucket sort, Address calculation, Sequential searching, Binary Searching, Index searching, Hash table methods.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Interpret and compute asymptotic notations of an algorithm to analyze the time complexity
CO2: Use of linear and non-linear data structures as the foundational base for computer solutions to problems
CO3: Demonstrate the ability to implement various types of static and dynamic lists
CO4: Implement binary trees, binary tree traversals, binary search trees and perform related analysis to solve problems
CO5: Implement various types of sorting algorithms

Text Books:

1. An Introduction to Data Structures with applications by J.P. Tremblay and P.G. Sorenson, Tata McGraw Hill.
2. Data structures, Algorithms ad Applications in C++ by Sartaj Sahni, WCB/McGraw Hill.
3. Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft,, Addison Wesley.

References Books:

1. Data Structures using C by Y. Langsam, M. J. Augenstein and A. M. Tenenbaum, Pearson Education.
2. Data Structures – A Pseudocode Approach with C by Richard F. Gilberg and Behrouz A. Forouzan, Thomson Brooks /COLE.

Course Name: **Computer Networks**
Course Code: **CS-302**
Course Type: **Institute Elective-I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the network models and architectures.
- To introduce the fundamental concepts relevant to performance of various routing protocols and design of new routing protocol.
- To enable the students to understand computers, software, networking technologies and information assurance to an organization's management, operations, and requirements.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introductory Concepts:** Goals and Applications of Networks, LAN, WAN, MAN, Wireless network, Network software: Protocol hierarchies, design issues of layers, Interfaces and services. Reference Model: The OSI reference model, TCP/IP reference model, Example networks: Novell Netware, The ARPANET, The Internet, X-25 Networks, network standards.	5L
UNIT-02	**Physical Layer:** Fourier Analysis, Maximum data rate of a channel, Transmission media, Wireless transmission, Narrowband ISDN, Broadband ISDN and ATM, Virtual circuits, Circuit switching, Communication satellite.	5L
UNIT-03	**Data Link Layer:** Data link layer design issues, services provided to network layers, Framing, Error control, Flow control, Error detection and correction, Elementary data link protocols, An unrestricted Simplex protocol, A Simplex Stop-and-Wait protocol, Simplex Protocol for a noisy channel, Sliding Window protocols, A one-bit Sliding protocol, A protocol using go-back-N, A protocol using selective repeat, Protocol specification and verification, Example data link protocol-HDLC, PPP and SLIP.	6L
UNIT-04	**Medium Access Sublayer:** Channel Allocations, Static and dynamic allocation in LAN and MAN, Multiple Access protocols, ALOHA, Carrier Sense multiple access protocols, WDMA protocols, Wireless protocols, Collision free protocols, Limited contention protocols, IEEE standard 802.3 and Ethernet, IEEE standard 802.4, Token bus IEEE standard 802.5, Token Ring, Distributed Queue Dual bus, Logical link control, bridges, High speed LAN, Satellite networks.	6L
UNIT-05	**Network Layer:** Network Layer design issue, Routing algorithms, Congestion Control Algorithms, Internetworking. **Transport Layer:** Transport services, Design issues, elements of transport protocols, simple transport protocols, Connection management, TCP, UDP.	8L
UNIT-06	**Session, Presentation and Application Layer:** Session Layer, Design issues, remote procedure call. Presentation Layer, Design issues, Data compression techniques, cryptography. Application Layer - File Transfer, Access and Management, Electronic mail, Virtual Terminals, Other applications, Example Networks - Internet and Public Networks.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1 Understand network models and architectures
CO2 Identify the pros and cons of choosing a suitable MAC layer protocol
CO3 Analyze the performance of various routing protocols and design of new routing protocol
CO4: Solve basic network design problems using knowledge of common local and wide area network architectures
CO5: Apply knowledge of computers, software, networking technologies and information assurance to an organization's management, operations, and requirements

Text Books:

1. Computer Networks by A.S. Tanenbaum, Prentice Hall of India.
2. Computer Networking: A Top-Down Approach Featuring the Internet by J. Kurose and K.W. Ross, Addison-Wesley.
3. Data and Computer Communication by W. Stallings, Prentice Hall of India.

Reference Books:

1. Network Programmability and Automation, Jason Edelman, 1st edition, O′Reilly.
2. Computer Networks A Top Down Approach, A Forouzan and F Mosharraf, Mc Graw Hill

Course Name: **Artificial Intelligence**
Course Code: **CS-303**
Course Type: **Institute Elective-I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the basic concepts related to Artificial Intelligence.
- To give understanding of the main abstractions and reasoning for intelligent systems.
- To enable the students to understand the basic principles of Artificial Intelligence in various applications

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Overview of AI problems, AI problems as NP, NP-Complete and NP Hard problems. Strong and weak, neat and scruffy, symbolic and sub-symbolic, knowledge-based and data-driven AI.	5 L
UNIT-02	**Search Strategies:** Problem spaces (states, goals and operators), problem solving by search, Heuristics and informed search, Minmax Search, Alpha-beta pruning. Constraint satisfaction (backtracking and local search methods).	8 L
UNIT-03	**Knowledge Representation and Reasoning:** propositional and predicate logic, Resolution and theorem proving, Temporal and spatial reasoning. Probabilistic reasoning, Bayes theorem. Totally-ordered and partially-ordered Planning. Goal stack planning, Nonlinear planning, Hierarchical planning.	9 L
UNIT-04	**Learning:** Learning from example, Learning by advice, Explanation based learning, Learning in problem solving, Classification, Inductive learning, Naive Bayesian Classifier, decision trees. **Natural Language Processing:** Language models, n-grams, Vector space models, Bag of words, Text classification. Information retrieval.	7 L
UNIT-05	**Agents:** Definition of agents, Agent architectures (e.g., reactive, layered, cognitive), Multi-agent systems- Collaborating agents, Competitive agents, Swarm systems and biologically inspired models. **Intelligent Systems:** Representing and Using Domain Knowledge, Expert System Shells, Explanation, Knowledge Acquisition.	7 L
UNIT-06	**Key Application Areas:** Expert system, decision support systems, Speech and vision, Natural language processing, Information Retrieval, Semantic Web.	5 L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Solve basic AI based problems.
CO2: Define the concept of Artificial Intelligence.
CO3: Apply AI techniques to real-world problems to develop intelligent systems.
CO4: Select appropriately from a range of techniques when implementing intelligent systems.

Text Books:

1. Artificial Intelligence by Elaine Rich, Kevin Knight and Shivashankar B Nair, Tata McGraw Hill.
2. Introduction to Artificial Intelligence and Expert Systems by Dan W. Patterson, Pearson Education.
3. Artificial Intelligence: A Modern Approach by S. Russell and P. Norvig, Prentice Hall.

References Books:

1. A First Course in Artificial Intelligence by Deepak Khemani, McGraw Hill Education.
2. Artificial Intelligence By Example: Acquire advanced AI, machine learning, and deep learning design skills, 2nd Edition.

Course Name: **Analysis & Design of Algorithms**
Course Code: **CS-311**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the asymptotic notations to analyze the performance of algorithms
- To introduce the fundamental concepts various problem-solving techniques such as divide and conquer, greedy algorithm, etc.
- To enable the students to understand the concepts of P, NP, NP-hard and NP-complete problems.

Unit Number	Course Content	Contact Hours
UNIT-01	**Algorithms Introduction:** Algorithm Design paradigms- motivation, concept of algorithmic efficiency, run time analysis of algorithms, Asymptotic Notations.	4L
UNIT-02	**Divide and Conquer Approach:** Structure of divide-and-conquer algorithms: sets and disjoint sets: Union and Find algorithms, quick sort, Finding the maximum and minimum, Quick Sort, Merge sort, Heap and heap sort. **Greedy Algorithms:** Optimal storage on tapes, Knapsack problem, Job sequencing with deadlines, Minimum Spanning trees: Prim's algorithm & Kruskal's algorithm, Huffman codes.	9L
UNIT-03	**Graph Algorithms:** Representation of graphs, BFS, DFS, Topological sort, strongly connected components; single source shortest paths: Bellman-Ford algorithm, Dijkstra's algorithm; All pairs shortest path: The Warshall's algorithm.	6L
UNIT-04	**Dynamic Programming:** Overview, difference between dynamic programming and divide and conquer, Matrix chain multiplication, Traveling Salesman Problem, Longest Common sequence, 0/1 knapsack. **Backtracking:** 8-Queen Problem, Sum of subsets, graph coloring, Hamiltonian cycles.	7L
UNIT-05	**Branch and Bound:** LC searching Bounding, FIFO branch and bound, LC branch and bound application: 0/1 Knapsack problem, Traveling Salesman Problem.	5L
UNIT-06	**Computational Complexity:** Complexity measures, Polynomial vs nonpolynomial time complexity; NP-hard and NP-complete classes, examples.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand asymptotic notations to analyze the performance of algorithms

CO2: Understand and apply various problem-solving techniques such as divide and conquer, greedy algorithm, dynamic programming, etc.

CO3: Solve given problem by selecting the appropriate algorithm design technique and justify the selection

CO4: Know the concepts of P, NP, NP-hard and NP-complete problems

Text Books:

1. Fundamentals of Computer Algorithms by E. Horowitz and S. Sahni, Galgotia.
2. Introduction to Algorithms by T.H. Cormen, C.E.Leiserson, R.L. Rivest, MIT Press, Cambridge.
3. The Design and Analysis of Computer Algorithms by A.V. Aho, J.E. Hopcroft and J.D. Ullman, Addison Wesley.

References Books:

1. Grokking Algorithms, Aditya Bhargava, 1st edition,Manning.
2. Data Structures and Algorithms Made Easy, Narasimha Karumanchi, 5th edition, Careermonk Publications

Course Name: **Compiler Design**
Course Code: **CS-312**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the different phases of compilation process
- To introduce the fundamental concepts relevant to compiler design techniques.
- To enable the students to understand the code optimization and code generation.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Compilers:** Compiler and translators need of translators, structure of a compiler, lexical analysis, syntax Analysis. Basic Parsing Techniques: Parsers, shift-reduce parsing, predictive parsing.	4L
UNIT-02	**Automatic Construction of Efficient Parsers:** LR parsers, canonical collection of LR(0) items, construction canonical LR parsing tables, construction LALR and SLR parsing tables using ambiguous grammars, an automatic parser generator, implementation of LR parsing tables, construction LALR sets of items.	9L
UNIT-03	**Syntax-Directed Translation:** Syntax directed translation schemes, implementation of syntax directed translation, intermediate code, postfix notation parse trees and syntax trees, three address code, quadruples and triples, translation of assignment statements, postfix translation with top down parser**.**	8L
UNIT-04	**Symbol Tables:** Contents of a table, data structures for symbol tables, representing scope information. Error detection and recovery: Errors, lexical-phase errors, syntax-phase errors, semantic errors.	7L
UNIT-05	**Introduction to Code optimization:** The principal source of optimization, loop optimization, DAG representation of basic blocks, value numbers and algebraic laws, global data flow analysis.	6L
UNIT-06	**Code Generation:** Object programs, problems in code generation, machine model, simple code generator, register allocation and assignment, code generation from DAG's, peephole optimization.	7L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand different phases of compilation process
CO2: Analyze and implement various parsing techniques
CO3: Understand and analyze intermediate code
CO4: Realize the importance of code optimization and code generation

Text Books:

1. Compilers Principles, Techniques and Tools by Alfred Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman, Pearson Education.
2. Compiler Design in C by Allen I. Holub, Prentice Hall of India.
3. Introduction to Compiler Techniques by J.P. Bennet, Tata McGraw-Hill.

References Books:

1. Practice and Principles of Compiler Building with C by Henk Alblas and Albert Nymeyer, Prentice Hall of India
2. Lex & Yacc by John R Levine Tony Mason and Doug Brown, O'reilly.
3. Engineering a Compiler by Keith D. Cooper and Linda Troczon, Morgan Kaufmann, Elsevier.

Course Name: **Computer Networks**
Course Code: **CS-313**
Course Type: **Discipline Core**

Contact Hours/Week: 3**L** Course Credits: **03**

Course Objectives

- To impart knowledge about the network models and architectures.
- To introduce the fundamental concepts relevant to performance of various routing protocols and design of new routing protocol.
- To enable the students to understand computers, software, networking technologies and information assurance to an organization's management, operations, and requirements.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introductory Concepts:** Goals and Applications of Networks, LAN, WAN, MAN, Wireless network, Network software: Protocol hierarchies, design issues of layers, Interfaces and services. Reference Model: The OSI reference model, TCP/IP reference model, Example networks: Novell Netware, The ARPANET, The Internet, X-25 Networks, network standards.	5L
UNIT-02	**Physical Layer:** Fourier Analysis, Maximum data rate of a channel, Transmission media, Wireless transmission, Narrowband ISDN, Broadband ISDN and ATM, Virtual circuits, Circuit switching, Communication satellite.	5L
UNIT-03	**Data Link Layer:** Data link layer design issues, services provided to network layers, Framing, Error control, Flow control, Error detection and correction, Elementary data link protocols, An unrestricted Simplex protocol, A Simplex Stop-and-Wait protocol, Simplex Protocol for a noisy channel, Sliding Window protocols, A one-bit Sliding protocol, A protocol using go-back-N, A protocol using selective repeat, Protocol specification and verification, Example data link protocol-HDLC, PPP and SLIP.	6L
UNIT-04	**Medium Access Sublayer:** Channel Allocations, Static and dynamic allocation in LAN and MAN, Multiple Access protocols, ALOHA, Carrier Sense multiple access protocols, WDMA protocols, Wireless protocols, Collision free protocols, Limited contention protocols, IEEE standard 802.3 and Ethernet, IEEE standard 802.4, Token bus IEEE standard 802.5, Token Ring, Distributed Queue Dual bus, Logical link control, bridges, High speed LAN, Satellite networks.	6L
UNIT-05	**Network Layer:** Network Layer design issue, Routing algorithms, Congestion Control Algorithms, Internetworking. **Transport Layer:** Transport services, Design issues, elements of transport protocols, simple transport protocols, Connection management, TCP, UDP.	8L
UNIT-06	**Session, Presentation and Application Layer:** Session Layer, Design issues, remote procedure call. Presentation Layer, Design issues, Data compression techniques, cryptography. Application Layer - File Transfer, Access and Management, Electronic mail, Virtual Terminals, Other applications, Example Networks - Internet and Public Networks.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1 Understand network models and architectures
CO2 Identify the pros and cons of choosing a suitable MAC layer protocol
CO3 Analyze the performance of various routing protocols and design of new routing protocol
CO4: Solve basic network design problems using knowledge of common local and wide area network architectures
CO5: Apply knowledge of computers, software, networking technologies and information assurance to an organization's management, operations, and requirements

Text Books:

1. Computer Networks by A.S. Tanenbaum, Prentice Hall of India.
2. Computer Networking: A Top-Down Approach Featuring the Internet by J. Kurose and K.W. Ross, Addison-Wesley.
3. Data and Computer Communication by W. Stallings, Prentice Hall of India.

Reference Books:

1. Network Programmability and Automation, Jason Edelman, 1st edition, O′Reilly.
2. Computer Networks A Top Down Approach, A Forouzan and F Mosharraf, Mc Graw Hill

Course Name: **Artificial Intelligence**
Course Code: **CS-314**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the basic concepts related to Artificial Intelligence.
- To give understanding of the main abstractions and reasoning for intelligent systems.
- To enable the students to understand the basic principles of Artificial Intelligence in various applications

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Overview of AI problems, AI problems as NP, NP-Complete and NP Hard problems. Strong and weak, neat and scruffy, symbolic and sub-symbolic, knowledge-based and data-driven AI.	5 L
UNIT-02	**Search Strategies:** Problem spaces (states, goals and operators), problem solving by search, Heuristics and informed search, Minmax Search, Alpha-beta pruning. Constraint satisfaction (backtracking and local search methods).	8 L
UNIT-03	**Knowledge representation and reasoning:** propositional and predicate logic, Resolution and theorem proving, Temporal and spatial reasoning. Probabilistic reasoning, Bayes theorem. Totally-ordered and partially-ordered Planning. Goal stack planning, Nonlinear planning, Hierarchical planning.	9 L
UNIT-04	**Learning:** Learning from example, Learning by advice, Explanation based learning, Learning in problem solving, Classification, Inductive learning, Naive Bayesian Classifier, decision trees. **Natural Language Processing:** Language models, n-grams, Vector space models, Bag of words, Text classification. Information retrieval.	7 L
UNIT-05	**Agents:** Definition of agents, Agent architectures (e.g., reactive, layered, cognitive), Multi-agent systems- Collaborating agents, Competitive agents, Swarm systems and biologically inspired models. **Intelligent Systems:** Representing and Using Domain Knowledge, Expert System Shells, Explanation, Knowledge Acquisition.	7 L
UNIT-06	**Key Application Areas:** Expert system, decision support systems, Speech and vision, Natural language processing, Information Retrieval, Semantic Web.	5 L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Solve basic AI based problems.
CO2: Define the concept of Artificial Intelligence.
CO3: Apply AI techniques to real-world problems to develop intelligent systems.
CO4: Select appropriately from a range of techniques when implementing intelligent systems.

Text Books:

1. Artificial Intelligence by Elaine Rich, Kevin Knight and Shivashankar B Nair, Tata McGraw Hill.
2. Introduction to Artificial Intelligence and Expert Systems by Dan W. Patterson, Pearson Education.
3. Artificial Intelligence: A Modern Approach by S. Russell and P. Norvig, Prentice Hall.

References Books:

1. A First Course in Artificial Intelligence by Deepak Khemani, McGraw Hill Education.
2. Artificial Intelligence By Example: Acquire advanced AI, machine learning, and deep learning design skills, 2nd Edition.

Course Name: **Advanced Operating System**
Course Code: **CS-351**
Course Type: **Discipline Elective-II**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To understand the Distributed System in details.
- To understand the logical clocks and physical clock.
- To understand the agreement, consensus and fault tolerance problems in distributed system.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Architectures of Distributed Systems, System Architecture types, issues in distributed operating systems, communication networks, communication primitives. Theoretical Foundations, inherent limitations of a distributed system, Lamport's logical clocks, vector clocks, casual ordering of messages, global state, cuts of a distributed computation – termination detection.	**8 L**
UNIT-02	**Distributed Mutual Exclusion:** introduction, the classification of mutual exclusion and associated algorithms, a comparative performance analysis. Distributed Deadlock Detection: Introduction, deadlock handling strategies in distributed systems, issues in deadlock detection and resolution, control organizations for distributed deadlock detection, centralized and distributed deadlock detection algorithms –hierarchical deadlock detection algorithms.	**8 L**
UNIT-03	**Agreement Protocols:** introduction, the system model, a classification of agreement problems, solutions to the Byzantine agreement problem, applications of agreement algorithms. Distributed resource management: introduction-architecture – mechanism for building distributed file systems – design issues – log structured file systems.	**8 L**
UNIT-04	**Distributed Shared Memory:** Architecture– algorithms for implementing DSM – memory coherence and protocols – design issues. Distributed Scheduling – introduction – issues in load distributing – components of a load distributing algorithm – stability – load distributing algorithm – performance comparison – selecting a suitable load sharing algorithm – requirements for load distributing -task migration and associated issues.	**8 L**
UNIT-05	**Failure Recovery and Fault Tolerance:** introduction– basic concepts – classification of failures – backward and forward error recovery, backward error recovery- recovery in concurrent systems – consistent set of check points –	**5 L**
UNIT-06	**Checkpointing:** Synchronous and asynchronous check pointing and recovery – check pointing for distributed database systems- recovery in replicated distributed databases.	**4 L**

Course Outcomes

CO1: Understand general issues of design and implementation of advanced modern operating systems.
CO2: Explain interprocess communication, distributed processing, sharing and replication of data and files in distributed systems.
CO3: Understand agreement protocols, distributed system architecture, and fault tolerance in distributed system.

Text Books:

1. Mukesh Singhal, Niranjan G. Shivaratri, "Advanced concepts in operating systems: Distributed, Database and multiprocessor operating systems", TMH, 2001.

2. Andrew S. Tanenbaum, "Modern operating system", PHI, 2003.

3. Pradeep K. Sinha, "Distributed operating system-Concepts and design", PHI, 2003.

References Books:

1. Operating Systems: Three Easy Pieces Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau Arpaci-Dusseau Books.

2. Distributed Systems: Concepts and Design by G. Coulouris,

Course Name: **Graph Theory**
Course Code: **CS-352**
Course Type: **Discipline Elective-II**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the graph data structures
- To enable the students to understand the basics of graph algorithms.
- To introduce the fundamental concepts relevant to understand the concepts of time and space complexity, worst case, average case and best-case complexities of graph algorithms.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Graphs, Isomorphism, Walks, Paths, Circuits, Trees, Properties of Trees, Cotrees and Fundamental Circuits, Cut Sets, Fundamental Cut Sets and Cut Vertices, Planar and Dual Graphs.	6 L
UNIT-02	**Graph Representation:** Metric Representation of Graphs, Coloring and covering and partitioning of a graph, chromatic number, chromatic partitioning, chromatic polynomials, matching, covering, four color problem, Directed graphs, some type of directed graphs, Directed paths, and connectedness,	6 L
UNIT-03	**Digraph Problems:** Euler digraphs, trees with directed edges, fundamental circuits in digraph, matrices A, B and C of digraphs adjacency matrix of a digraph, enumeration, types of enumeration, counting of labeled and unlabeled trees, polya's theorem, graph enumeration with polya's theorem;	6 L
UNIT-04	**Graph Algorithms:** Elementary Graph 12 Algorithms, Representations of graphs, Breadth-first search, Depth-first search, Topological sort, strongly connected components; Minimum Spanning Trees: Growing a minimum spanning tree, The algorithms of Kruskal and Prim,	6 L
UNIT-05	**Shortest Paths:** Shortest paths and relaxation, Dijkstra's algorithm, The Bellman-Ford algorithm, Single source shortest paths in directed acyclic graphs, Difference constraints and shortest paths, All-Pairs Shortest Paths: Shortest paths and matrix multiplication, The Floyd-Warshall algorithm, Johnson's algorithm for sparse graphs, and A general framework for solving path problems in directed graphs.	5 L
UNIT-06	**Network Flow:** Maximum Flow, Flow networks, The Ford-Fulkerson method, Maximum bipartite matching, Preflow-push algorithms, The lift-to-front algorithm. Special topics in graph theory and network algorithms.	6 L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand graph problems and implement effective solutions.

CO2: Understand asymptotic notations to analyse the performance of graph algorithms.

Text Books:

1. Introduction to Algorithms by T. H. Cormen, C. E. Leiserson and R. L. Rivest, Prentice Hall of India.
2. Graph Theory with Applications to Engineering and Computer Science by N. Deo,Prentice Hall of India.
3. Introduction to Graph Theory by D. B. West, Prentice Hall of India.

References Books:

1. Advanced Graph Theory by R. Diestel, Springer Verlag Heidelberg, New York.
2. Algorithm Design: Foundations, Analysis, and Internet Examples by M. T. Goodrich and R. Tamassia, Wiley.

Course Name: **Information Retrieval**
Course Code: **CS-353**
Course Type: **Discipline Elective-II**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To learn concepts of Information Retrieval from documents and web portals.
- To evaluate the efficiency of existing Information Retrieval approaches.
- To implement Information Retrieval approaches in real applications.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Introductory lecture, discussion of areas and applications of Information Retrieval, Basic Concepts, Data Representation,	**5L**
UNIT-02	**Foundations of Information Retrieval:** Boolean retrieval model, Dictionary and Postings: Tokenization, Stop words, Stemming, Inverted index, Skip pointers, Phrase queries, Tolerant Retrieval: Wild card queries, Permuterm index, Bigram index, Spelling correction, Edit distance, Jaccard coefficient, Soundex, Evaluation Precision, Recall, F-measure, Normalized recall.	**5L**
UNIT-03	**Retrieval Models:** Vector Space Model, Query-document scoring using Vector space model, Language Model, Latent Semantic Indexing. learning-based ranking algorithms. Latent Semantic Indexing, Eigen vectors, Singular value decomposition.	**5L**
UNIT-04	**Probabilistic Information Retrieval:** Probabilistic relevance feedback, Probability ranking principle, Binary Independence Model, Bayesian network for text retrieval, Structural terms, Probabilistic Retrieval Models, probabilistic topic models, Query Expansion Relevance feedback, Rocchio algorithm, Probabilistic relevance feedback.	**8L**
UNIT-05	**Multimedia Information Retrieval:** Content Based Image retrieval, Challenges in Image retrieval, Image representation, Indexing and retrieving images, Relevance feedback, Text Categorization, Text Clustering, NLP in IR, Sentiment Analysis, Document classification	**6L**
UNIT-06	**Advanced Topics in Web Information Retrieval:** The structure of the web, Queries and users, Static ranking, Dynamic ranking, evaluating web search, the user, paid placement, Search engine optimization/spam, Web Search Architectures: Crawling, Meta-crawlers and Focused Crawling, Web Crawlers: Crawling the web, Document feeds, storing documents and detecting duplicates.	**6L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand different Information Retrieval models.
CO2: Evaluate various Information Retrieval models.
CO3: Evaluate and optimize search engines and develop optimized web sites.

Text Books:

1. Introduction to Information Retrieval by Manning, Raghavan and Schütze.
2. Modern Information Retrieval by R. Baeza-Yates and B. Ribeiro-Neto.
3. Foundations of Statistical Natural Language Processing by Manning and Schutze, MIT Press, 1999.

References Books:

1. Jurafsky and Martin, Speech and Language Processing, Pearson; 2nd edition 2008.
2. Search Engines Information Retrieval in Practice by W. Bruce Croft, Donald Metzler and Trevor Strohman, Pearson Education, Inc. 2015.

Course Name:	**Compiler Design Lab**
Course Code:	**CS-315**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To enlighten the student with knowledge base in compiler design and its applications
- To implement Lexical Analyzer using Lex tool & Syntax Analyzer or parser using YACC Tool
- To implement front end of the compiler by means of generating Intermediate codes.
- To implement code optimization techniques

List of Experiments

1. Write a C/C++ program to search the number of occurrences of any word, string and sub sequence in a given string
2. Write a C/C++ program to find the number of occurrences of any word in a file and replace that with another word.
3. Implement the lexical analyzer using JLex, Flex or other lexical analyzer generating tools and write the following program using LEX.
    - Program to count the number of characters, words, spaces and lines in a given input file.
    - Program to count the number of comment lines in a given C program. Also eliminate them and copy the resulting program into separate file.
    - Program to recognize a valid arithmetic expression and to recognize the identifiers and operators present. Print them separately.
    - Program to recognize and count the number of identifiers in a given input file.
4. Write a program to check whether a string belongs to the grammar or not. Write a program to generate a parse tree. Write a program to check whether a grammar is left recursive and left factoring. Write a program to remove left recursion and left factoring. Write a program to implementation FIRST and FOLLOW for the given grammar.
5. Write a program to implementation to implement recursive-Descent and Non-recursive Predictive parser. Write a program to check whether a given grammar is LL(1) or not.
6. Write a C/C+ + program to implementation of shift reduce parsing Algorithm. Write a C/C++ program to construct LR Parsing table and Implement LR parsing algorithm. Write a C/C++ program to construct CLR Parsing table and implement CLR parsing algorithm
7. Write a C/C++ program to construct LALR Parsing table and implement LALR parsing algorithm. YACC program to recognize a valid arithmetic expression that uses operators +, -,* and /. YACC program to recognize strings ‘aabb’,’abbb’, ‘ab’ and ‘a’ using the grammar (a^nb^n,n>=10), where a and b are input symbols of the grammar.
8. Write a C program to implement the syntax-directed definition of “if E then S1” and “if E then S1 else S2”,
9. Write a C/C++ program to implementation a Symbol Table with function to create, insert, modify, search and display.

Note: The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Design Lexical analyzer for given language using C and LEX tools.
CO2: Design and convert BNF rules into YACC form to generate various parsers.
CO3: Generate machine code from the intermediate code forms.
CO4: Implement Symbol table and compiler design aspects.

Course Name:	**Computer Networks Lab**
Course Code:	**CS-316**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To create client and server applications using the "Sockets" API and the implementation of Data link layer protocol and TCP layer
- To conduct computer communication network simulations.
- To have a hands on experience of computer network simulation and modeling techniques using NS-3 simulation software.

List of Experiments

1. To become familiar with networking accessories and facilities in the Department of Computer Science and Engineering:
   - Find out what networking devices are installed in the department
   - Describe the network type and topology of the department
   - File and printer sharing in different OSs
   - Network address configuration in different OSs
   - Finding IP and MAC address in different OSs
   - Workgroup and domain configuration
   - Use of utilities: arp, ipconfig/ifconfig, tracert, ns lookup
2. Examine packets flow across a network segment and see the operation of various Internet protocols across the different layers in TCP/IP stack. (Hint: Use utilities netstat, snoop, tcp dump, …)
3. Use Unix sockets to implement a simple client and server that communicate over the network. (Reference: Unix Network Programming by W. Richard Stevens, Prentice Hall)
4. Simulate various multiple access protocols (Aloha, slotted Aloha, p-persistent and non-persistent) and compare their performance at different loads.
5. Write a program to display the IP address and MAC address of a machine.
6. Implement ARP and RARP protocols for CSE LAN.
7. Install Ethereal on a computer. Set Ethereal to capture with a filter option of your choice. Load a webpage or send an email to a friend and stop capturing. Analyze the packets. See if you can read any or all of the data transmitted. Write down your findings.
8. Write a program to simulate routing using flooding. Each packet should contain a counter that is decremented on each hop. When the counter gets to zero, the packet is discarded. Time is discrete, with each line handling one packet per time interval. Make three versions of the program: all lines are flooded, all lines except the input line are flooded, and only the (statically chosen) best k lines are flooded. Compare flooding with deterministic routing (k = 1) in terms of both delay and the bandwidth used.
9. Write a program that simulates a computer network using discrete time. The first packet on each router queue makes one hop per time interval. Each router has only a finite number of buffers. If a packet arrives and there is no room for it, it is discarded and not transmitted. Instead, there is an end-to-end protocol, complete with timeouts and acknowledgement packets, that eventually regenerates the packet from the source router. Plot the throughput of the network as a function of the end-to-end timeout interval, parameterized by error rate.
10. Design and implement a chat system that allows multiple groups of users to chat. A chat coordinator resides at a well-known network address, uses UDP for communication with chat clients, sets up chat servers for each chat session, and maintains a chat session directory. There is one chat server per chat session. A chat server uses TCP for communication with clients. A chat client allows users to start, join, and leave a chat session. Design and implement coordinator, server, and client code.

Note: The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Ability to invoke analytical studies of Computer Networks through network simulation.

CO2: Technical knowhow of the various components in NS-3 toolkit and its importance in designing a real network.

Course Name: **Digital Image Processing**
Course Code: **CS-321**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the fundamental steps of digital image processing.
- To introduce the fundamental concepts relevant to digital image processing.
- To enable the students to understand the various image compression techniques, pattern recognition and classification techniques.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Digital image representation, Fundamental steps in Digital image processing, Elements of Digital Image processing systems, Elements of visual perception, Image model, Sampling and quantization, Relationship between pixels, Imaging geometry. Basic Mathematical Tools Used in Digital Image Processing.	5L
UNIT-02	**Image Enhancement:** Enhancement by point processing, Sample intensity transformation, Histogram processing, Image subtraction, Image averaging, Spatial filtering, Smoothing filters, Sharpening filters, Frequency domain: Low-Pass, High-Pass, Homomorphic filtering. Image Restoration and Reconstruction, Color Image Processing	9L
UNIT-03	**Image Compression:** Coding redundancy, Inter-pixel redundancy, fidelity criteria, Image compression models, Error-free compression, Variable length coding, Bit-plane coding, Loss-less predicative coding, Lossy compression, Image compression standards, Fractal Compression, Real-Time image transmission, JPEG and MPEG.	8L
UNIT-04	**Image Segmentation:** Detection of discontinuities, Edge linking and boundary detection, Thresholding, Region Based Segmentation, Use of motion in segmentation, Spatial techniques, Frequency domain techniques.	6L
UNIT-05	**Spatial Operations and Transformations:** Spatially dependent transform template and convolution, Window operations, 2- Dimensional geometric transformations.	5L
UNIT-06	**Pattern Recognition:** Classification and description, Structure of a pattern recognition system, feature extraction, Classifiers, Decision regions and boundaries, discriminate functions, Supervised and Unsupervised learning, PR-Approaches statistics, syntactic and neural. **Statistical Pattern Recognition:** Statistical PR, Classifier Gaussian Model, Classifier performance, Risk and error, Maximum likelihood estimation, Bayesian parameter estimation approach, clustering for unsupervised learning and classifiers.	9L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand fundamental steps of digital image processing.
CO2: Understand and implement image enhancement techniques.
CO3: Implement and compare various image compression techniques.
CO4: Understand and implement pattern recognition and classification techniques.

Text Books

1. Digital Image Processing by Rafael C. Gonzalez and Richard E. Woods, Pearson Education.
2. Introductory Computer Vision and Image Processing by Andrian Low, McGraw Hill.
3. Pattern Recognition-Statistical, Structural and neural approach by Robert Schalkoff, John Willey &Sons.

References Books

1. Digital Image Processing by W.K. Pratt, McGraw Hill.
2. Fundamentals of Image Processing by A. K. Jain, Pearson

Course Name: **Database Management Systems**
Course Code: **CS-322**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the DBMS concept, data models and architecture
- To introduce the fundamental concepts relevant to ER model and its mapping to relational model, relational algebra and SQP.
- To enable the students to understand the factors that cause the concurrency and recovery strategies for DBMS.

Unit Number	Course Content	Contact Hours
UNIT-01	**Basic Concepts:** Introduction to File and Database systems- Database system structure – concepts and architecture, date models, schemas & instances, DBMS architecture & data independence, database languages & interfaces, Data Model, ER model.	4L
UNIT-02	**Relational Models:** SQL – Data definition- Queries in SQL-relational model concepts, relational model constraints, relational algebra, SQL- a relational database language: date definition in SQL, view and queries in SQL, specifying constraints and indexes in SQL; relational database management systems-Updates, Views, Integrity and Security.	5L
UNIT-03	**Relational Database design**: Functional dependences and Normalization for Relational Databases, normal forms based on primary keys, (1NF, 2NF, 3NF & BCNF), lossless join and dependency preserving decomposition, converting ER-diagrams into relations**.**	5L
UNIT-04	**Data Storage and query Processing:** Record storage and Primary file organization- Secondary storage Devices, Operations on Files, Heap File, Sorted Files, Hashing Techniques, Index Structure for files, Different types of Indexes- B-Tree - B+Tree, Query Processing.	9L
UNIT-05	**Transaction Management:** Transaction Processing, Need for Concurrency control, Desirable properties of Transaction, Schedule and Recoverability, Serializability and Schedules; Concurrency Control, Types of Locks, Two Phases locking, Deadlock, Time stamp based concurrency control, Recovery Techniques, Concepts- Immediate Update- Deferred Update, Shadow Paging.	9L
UNIT-06	**Current Trends:** Introduction to Distributed and parallel databases, Deductive Databases, Multimedia Databases, Real-Time Databases.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand DBMS concept, data models and architecture
CO2: Understand ER model and its mapping to relational model
CO3: Use of relational algebra and SQP
CO4: Apply normalization to build database
CO5: Understand concurrency and recovery strategies for DBMS

Text Books:

1. An Introduction to Database Systems by C.J.Date, Addison Wesley.
2. Fundamentals of Database Systems by Elmsari and Navathe, Addison Wesley.
3. Database System Concepts by Abraham Silberschatz, Henry F. Korth and S. Sudarshan, McGraw-Hill
4. Fundamental Database Systems by Ramez Elmasri and Shamkant B. Navathe, Pearson Education.

References Books:

1. Database System Implementation by Hector Garcia–Molina, Jeffrey D.Ullman and Jennifer Widom, Pearson Education.
2. Database System, Design, Implementation and Management by Peter Rob and Corlos Coronel, Thompson Learning Course Technology.

Course Name: **Software Engineering**
Course Code: **CS-323**
Course Type: **Discipline Core**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To impart knowledge about the concept of software development and software engineering
- To introduce the fundamental concepts relevant to comprehend different software engineering process models.
- To enable the students to understand the factors that affect the design of software projects and do the cost estimation.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Problem domain, software engineering challenges, software engineering approach. Software Processes: Software process, characteristics of software process, software development process models, other processes. Software Requirements Analysis and Specification: Software requirements, problem analysis, requirements specification, functional specification with use cases, validation, matrices.	8L
UNIT-02	**Software Architecture:** Role of software architect, architecture views, component and connector view, architecture style for C & C view, discussion and evaluating architectures. Planning a software project: Effort estimation, project scheduling and staffing, software configuration management plan, quality assurance plan, risk management, project monitoring plan.	6L
UNIT-03	**Function Oriented Design:** Design principles, module level concepts, design notation and specification, structured design methodology, verification, metrics. Object oriented design: OO concepts, design concept, Unified Modeling Language, design methodology, metrics. **Detailed Design, Software Measurements, Metrics and Models:** Detailed design and PDL, verification, Metrics and their scope, Qualities of a good Software metrics, classification of metrics, Cost estimation models COCOMO, Quality attributes, SQA, Quality Standards, ISO 9000 and CMM.	8L
UNIT-04	**Coding:** Programming principles and guidelines, coding process, refactoring, verification, metrics. Testing: Testing fundamentals, black-box testing, white-box testing, testing process, defect analysis and prevention, metrics - reliability estimation. CASE Tools: Types of CASE tools, advantages and components of CASE tools, Unified Modeling Language (UML). **IPR and Copyright:** Introduction and the need for intellectual property right (IPR), IPR in India – Genesis and Development, Patents: Searching, Drafting, Filling and Granting of Patents; Copyrights: Coverage and Need; Trademarks: Types, Rights, Kinds of signs of Trademarks; Geographical Indications.	9L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand and analyze the concept of software development and software engineering
CO2: Compare and comprehend different software engineering process models
CO3: Design of software projects and do the cost estimation
CO4: Apply different software testing techniques

Text Books:

1. An Integrated Approach to Software Engineering by Pankaj Jalote, Narosa Publishing.
2. Software Engineering: A Practitioner's Approach by Pressman Roger R, Tata McGraw Hill.
3. Fundamentals of Software Engineering by Rajib Mall, Prentice Hall of India.

References Books:

1. UML Bible by Tom Pender, Wiley Dreamtech.
2. Software Engineering by Ian Sommerville, Addison-Wesley

Course Name: **Game Theory**
Course Code: **CS-341**
Course Type: **Discipline Elective-III**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To teach students some strategic considerations to take into account making their choices.
- To learn basic concepts of game theory.

Unit Number	Course Content	Contact Hours
UNIT-01	**Basic Solution concepts and computational issues:** Games, Old and New; Games Strategies, Costs and Payoff, Basic Solution Concepts; Finding Equilibria and Learning in Games.	8 L
UNIT-02	**Refinement of Nash:** Games with Turns and Sub game, Perfect Equilibrium: Cooperative games, markets and their Algorithmic Issues.	5L
UNIT-03	**The Complexity of finding Nash Equilibria:** Introduction, Lemke Howson algorithm, succinct representation of games. Graphical Games: Computing Nash Equilibria in Tree Graphical Games, Graphical Games and correlated Equilibria.	8L
UNIT-04	**Cryptography and Game theory:** Cryptographic notation and settings, game theory notation and settings, cryptographic influence on game theory and Game theoretic influence on cryptography.	6 L
UNIT-05	**Distributed algorithmic mechanism design:** two examples of DAMD, Interdomain routing Cost sharing. Incentive and Pricing in Communication Networks Large Network Competitive model,	8 L
UNIT-06	**Pricing and Resource allocation:** Game theoretic model Incentive and Information security, Misaligned incentive, Informational Asymmetries, Complex network and topology.	6 L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Solve strategic games between two and more agents in non-cooperative scenario.
CO2: Analyze and solve both simultaneous-moves and sequential-moves games.
CO3: Learn different methods to solve games

Text Books:

1. A Course in Game Theory by M. J. Osborne and A. Rubinstein, MIT Press.
2. An Introduction to Game Theory by M. J. Osborne, Oxford University Press.
3. Algorithmic Game Theory by N. Nisan, T. Rougharden, E. Tardos and V. V. Vazirani, Cambridge University Press.

References Books:

1. Fun and Games: A Text on Game theory by K. Binmore, AIBS publisher

2. Stability and Perfection of Nash Equilibria by Eric van Damme, Springer Berlin, Heidelberg.

3. Game Theory: Analysis of Conflict by Roger Myerson, Harvard University Press.

Course Name: **Computer Vision**
Course Code: **CS-342**
Course Type: **Discipline Elective-III**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To study the development of algorithms and techniques to analyze and interpret the visible world around us.
- Be familiar with both the theoretical and practical aspects of computing with images.
- To understand the basic concepts of Computer Vision.
- Understand the geometric relationships between 2D images and the 3D world.
- Ability to apply the various concepts of Computer Vision in other application areas.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to computer vision and its applications:** Digital image formation and low-level processing: Fundamentals of Image Formation, Transformation: Orthogonal, Euclidean, Affine, Projective, etc;	6L
UNIT-02	**Digital Image Fundamentals:** Image Enhancement in Spatial Domain; Gray Level Transformation, Histogram Processing, Spatial Filters. Image Transforms: Fourier Transform and their properties, Fast Fourier Transform, Other Transforms	6L
UNIT-03	**Feature Extraction:** Edges - Canny, LOG, DOG; Line detectors (Hough Transform), Corners - Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, ScaleSpace Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT.	8L
UNIT-04	**Image Segmentation:** Region Growing, Edge Based approaches to segmentation, Graph-Cut, Mean-Shift, MRFs, Texture Segmentation; Object detection.	6L
UNIT-05	**Pattern Analysis:** Basics of Probability and Statistics, Clustering: K-Means, K-Medoids, Mixture of Gaussians, Classification: Discriminant Function, Supervised, Un-supervised, Semi-supervised; Classifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA, ICA; Non-parametric methods	6L
UNIT-06	**Applications:** Image Classification, feature based methods, deep networks, Object Detection, Semantic Segmentation.	4L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the fundamental problems of computer vision.
CO2: Implement various techniques and algorithms used in computer vision.
CO3: Analyze and evaluate critically the building and integration of computer vision algorithms.
CO4: Demonstrate awareness of the current key research issues in computer vision.

Text Books:

1. Computer Vision Algorithms and Applications, Richard Szeliski, Springer.
2. Computer Vision: A Modern Approach Hardcover, David Forsyth and Jean Ponce, Pearson.
3. Digital Image Processing by R. Gonzalez and r. E. Wood, Prentice Hall of India.
4. Introductory Computer Vision and Image Procession by Andrian low, McGraw Hill CO.

References Books:

1. Digital Image Processing by W.K. Pratt, McGraw Hill.
2. Computer Vision: Models, Learning, and Inference by Simon J. D. Prince, Cambridge University Press.

Course Name: **Natural Language Processing**
Course Code: **CS-343**
Course Type: **Discipline Elective-III**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To understand the application of computational methods in linguists.
- To apply statistical and probabilistic methods for parameter estimation and inference
- To know how the computational methods, give insight into observed human language phenomena.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Knowledge in speech and language processing, Ambiguity, Models and Algorithms, Brief History, NLP and its Applications, Regular Expressions and Finite-State Automata. Morphology and Transducers: Inflectional and derivational morphology, finite state morphological parsing, Combining FST Lexicon and rules. Lexicon free FST: Porter Stemmer	7L
UNIT-02	**Word Classes and Part-of-Speech Tagging:** English word classes, Targets for English, Part of speech Tagging, Rule Based part of speech Tagging, Transformation Based Tagging. Context Free Grammars for English: Constituency, Context Free rules and Trees, Sentence level construction, The Noun Phrase, Coordination, Agreement, The verb phrase and sub-categorization.	7L
UNIT-03	**N-grams:** Counting Words in Corpora, Simple (Unsmoothed) N-Grams: Counting words in Corpora, N-Gram probabilities, Training and Test sets, Evaluating N-Gram Perplexity.	6L
UNIT-04	**Smoothing Parsing with context free grammars:** Parsing as Search, Basic Top down Parser, Problems with basic top-down-parsers, the early Algorithm, Finite state parsing method.	6L
UNIT-05	**Named Entity Recognition:** Semantics-Meaning representation, semantic analysis, lexical semantics, WordNet, Word Sense Disambiguation. **Pragmatics:** Discourse, Reference Resolution, Text Coherence, Discourse structure	5L
UNIT-06	**Advanced Topics:** Machine Translation and Performance Metrics: Machine Translation issues, MT Evaluation, automatic evaluation BLEU, NIST, METEOR, ORANGE, Information Retrieval: Vector space model, term weighting, homonymy, polysemy, synonymy, improving user queries.	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Compare and contrast natural language processing approaches.
CO2: Comprehend and analyze various elements of speech processing.
CO3: Design and develop machine learning techniques in the area of NLP.

Text Books:

1. Speech and Language Processing by Daniel Jurafsky and James H. Martin, Prentice Hall
2. NLP: A Paninian Perspective by Akshar Bharati, Vineet Chaitanya, and Rajeev Sangal, Prentice Hall, New Delhi.
3. Language as a Cognitive Process by T. Winograd, Addison-Wesley.

Reference Books:

1. Natural Language Understanding, James Allen, The Benjamin/ Cummins Publishing Company.
2. Handbook of Natural Language Processing, Nitin Indurkhya and Fred J. Damerau, CRC Press.
3. Natural language Processing with Python, Steven Bird, Ewan Klein and Edward Lopar, O'Reilly.

Course Name: **Cloud Computing**
Course Code: **CS-361**
Course Type: **Discipline Elective-IV**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To provide comprehensive knowledge of fundamental concepts of cloud computing
- To demonstrate an understanding of Service models, deployment models, Virtualization.
- To describe the programming and software environments of Cloud.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Cloud Computing:** Roots of Cloud Computing, Layers and Types of Clouds, Features of a Cloud, Cloud Infrastructure Management, Infrastructure as a Service Providers, Platform as a Service Providers, and Challenges and Opportunities**.**	5L
UNIT-02	**Virtualization and Resource Provisioning in Clouds:** Introduction and Inspiration, Virtual Machines (VM), VM Provisioning and Manageability, VM Migration Services, VM Provisioning in the Cloud Context, and Future Research Directions.	7L
UNIT-03	**Cloud Computing Architecture:** Cloud Benefits and Challenges, Market-Oriented Cloud Architecture, SLA-oriented Resource Allocation, Global Cloud Exchange; Emerging Cloud Platforms, Federation of Clouds	7L
UNIT-04	**Emerging Cloud Computing Platforms:** Comparison of Cloud Platforms (e.g., AWS, Azure, Google Cloud), Federation of Clouds, Interoperability between Cloud Platforms, Hybrid Cloud Deployment Models, Cloud Security and Compliance Requirements	5L
UNIT-05	**Programming Enterprise Clouds using Aneka:** Introduction, Aneka Architecture, Aneka Deployment, Parallel Programming Models, Thread Programming using Aneka, Task Programming using Aneka, and MapReduce Programming using Aneka, Parallel Algorithms, Parallel Data mining, Parallel Mandelbrot, and Image Processing.	6L
UNIT-06	**Advanced Topics and Cloud Applications:** Integration of Private and Public Clouds, Cloud Best Practices, GrepTheWeb on Amazon Cloud, ECG Data Analysis on Cloud using Aneka, Hosting Massively Multiplayer Games on Cloud, and Content Delivery Networks Using Clouds, and Hosting Twitter and Facebook on Cloud.	6L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Identify roots of cloud computing and fundamental concepts of cloud computing.
CO2: Describe Programming Enterprise Clouds using Aneka.
CO3: Apply principles of programming and software environments of Cloud.

Text Books

1. Cloud Computing: Principles and Paradigms by R. Buyya, J. Broberg and A. Goscinski, Wiley Press,
2. Cloud Computing: A Practical Approach by A.T. Velte, T. J. Velte, Robert Elsenpeter, McGraw-Hill.
3. Cloud Computing Bible by B. Sosinsky, Wiley Publishing.

References Books

1. Architecting the Cloud: Design Decisions for Cloud Computing Service Models by M. J. Kavis, Wiley Publishing.
2. Cloud Computing: Concepts, Technology & Architecture, Thomas Erl , Ricardo Puttini and Zaigham Mahmood, Pearson.
3. Architecting the Cloud: Design Decisions for Cloud Computing Service Models, Michael J. Kavis. Wiley.

Course Name: **Statistical Computing**
Course Code: **CS-362**
Course Type: **Discipline Elective-IV**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To learn the probability distributions and density estimations to perform analysis of various kinds of data.
- To explore the statistical analysis techniques using Python and R programming languages.

Unit Number	Course Content	Contact Hours
UNIT-01	**The learning problem:** Risk functions, Well-posed and ill-posed problems; Supervised Learning Vs Function Approximation; Bias Variance Trade-off; Curse of Dimensionality.	**6 L**
UNIT-02	**Probability Theory:** Sample Spaces, Events, Axioms, Counting, Conditional Probability and Bayes' Theorem, The Binomial Theorem, Random variable and distributions, Mean and Variance of a Random Variable, Binomial-Poisson-Exponential and Normal distributions, Curve Fitting and Principles of Least Squares, Regression and correlation.	**7L**
UNIT-03	**Introduction to R**: Packages, Scientific Calculator, Inspecting Variables, Vectors, Matrices and Arrays, Lists and Data Frames, Functions, Strings and Factors, Flow Control and Loops, Advanced Looping, Date and Times.	**6L**
UNIT-04	**Introduction to Python**: Packages, Fundamentals of Python, Inserting and Exporting Data, Data Cleansing Checking and Filling Missing Data, Merging Data, Operations, Joins.	**5L**
UNIT-05	**Sampling Distributions & Descriptive Statistics:** The Central Limit Theorem, distributions of the sample mean and the sample variance for a normal population, Test of Hypothesis, Testing for Attributes, Mean of Normal Population, One-tailed and two-tailed tests.	**6 L**
UNIT-06	**Applications:** Least Squares vs Nearest Neighbour, PCA, LDA, Perceptron, Decision Boundaries, Regression Analysis, Logistic Regression, Introduction to Neural Networks.	**6L**

Course Outcomes

CO1: Implement statistical analysis techniques for solving practical problems.
CO2: Perform statistical analysis on variety of data.
CO3: Apply principles and algorithms to evaluate models generated from data.

Text Books:

1. Dalgaard, Peter, "Introductory statistics with R", Springer Science & Business Media, 2008.

2. Samir Madhavan, "Mastering Python for Data Science", Packt, 2015.

3. Sheldon Ross, "Introduction to Probability and Statistics for Engineers and Scientists", 4th edition, Academic Press, 2009.

Reference Books:

1. Paul Teetor, "R Cookbook, O'Reilly, 2011.

Course Name: **Neural Network**
Course Code: **CS-363**
Course Type: **Discipline Elective-IV**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the basic principles, techniques, and applications of neural network.
- Provide the mathematical background for carrying out the optimization associated with neural network learning
- Develop the skills to gain basic understanding of the areas of Artificial Neural Networks

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to ANNs:** Biological neural networks, Pattern recognition tasks, Computational models of neurons, MP Neurons, Introduction to perceptron's, Perceptron learning algorithm, Multilayer network of perceptron.	**8L**
UNIT-02	**Learning Principles:** Feed forward neural networks, Multilayer feed forward neural networks (MLFFNNs), Backpropagation, Gradient Descent	**6L**
UNIT-03	**Traditional NNs:** Auto associative neural networks, Other types of MLFFNNs, PCA, Autoencoders, Sparse and contrastive autoencoders.	**6L**
UNIT-04	**Advanced NNs:** Convolutional Neural networks, Visualizing CNNs, GradCAM, Guided Backpropagation,	**8L**
UNIT-05	**Theoretical Aspects:** Bias-Variance trade-off, Regularization theory, L2 regularization, Early Stopping, Dataset Augmentation, Ensemble Methods, Activation functions, Weight initialization, Batch normalization.	**8L**
UNIT-06	**Applications of ANN:** Pattern classification – Recognition of Olympic games symbols, Recognition of printed Characters. Neocognitron – Recognition of handwritten characters. Convert English text to speech. Recognition of consonant vowel (CV) segments, texture classification and segmentation	**6L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand the fundamental theory and concepts of neural networks.
CO2: Identify different neural network architectures and algorithms.
CO3: To learn how to design and how to supervised and unsupervised artificial neural networks
CO4: Know some application of artificial neural networks and their limitations.

Text Books:

1. Yegnanarayana, Artifical Neural Networks, Prentice Hall of India,1999
2. Satish Kumar, Neural Networks – A Classroom Approach, Tata Mc Graw-Hill, 2003
3. S. Haykin, Neural Networks – A Comprehensive Foundation, Prentice Hall, 1998

References Text:

1. Neural Networks for Pattern Recognition, Christopher M. Bishop, Oxford University Press.
2. Neural Networks and Deep Learning: A Textbook Hardcover, Charu C. Aggarwal, Spriger.
3. Deep Learning, Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press

Course Name: **Distributed Systems**
Course Code: **CS-381**
Course Type: **Stream Core-I**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To impart knowledge about the concepts of various architectural and fundamental models in distributed systems.
- To introduce the fundamental concepts relevant to client server and peer to peer interactions between clients and sever.
- To enable the students to understand the factors that cause the intricacies involved in distributed transactions.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Characterization of Distributed System, Examples of distributed systems, Resource sharing, challenges. **System Models:** Architectural Models: Client-Server Models, Proxy Server Models, Peer-to-Peer Models, Thin Clients, Mobile Codes, Mobile Agents, Fundamental Models: Interaction Model, Failure Model and Security Model. Global State: Clock Synchronization, Clock Synchronization Algorithms: synchronization in synchronous systems, Cristian's algorithm, Berkley Algorithm, NTP and its modes, Lamport's logical clocks, vector clocks, casual ordering of messages, global state determination, cuts, distributed debugging.	8L
UNIT-02	**Distributed Mutual Exclusion:** introduction, the classification of mutual exclusion and associated algorithms, a comparative performance analysis. Distributed Deadlock Detection: Introduction, deadlock handling strategies in distributed systems, issues in deadlock detection and resolution, control organizations for distributed deadlock detection, centralized and distributed deadlock detection algorithms –hierarchical deadlock detection algorithms.	8L
UNIT-03	**Agreement protocols:** introduction, the system model, a classification of agreement problems, solutions to the Byzantine agreement problem, applications of agreement algorithms. Distributed resource management: introduction-architecture – mechanism for building distributed file systems – design issues – log structured file systems.	8L
UNIT-04	**Distributed shared memory-Architecture–** algorithms for implementing DSM – memory coherence and protocols – design issues. Distributed Scheduling – introduction – issues in load distributing – components of a load distributing algorithm – stability – load distributing algorithm – performance comparison – selecting a suitable load sharing algorithm – requirements for load distributing -task migration and associated issues.	7L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand concepts of various architectural and fundamental models in distributed systems
CO2: Implement client server and peer to peer interactions between clients and sever
CO3: Analyze time synchronization protocols and select a suitable time synchronization protocol
CO4: Analyze and evaluate various election and agreement protocols in distributed environment
CO5: Analyze various intricacies involved in distributed transactions

Text Books

1. Distributed Systems: Concepts and Design by G. Coulouris, J. Dollimore, and T. Kindberg, Pearson Education.
2. Distributed Systems: Principles and Paradigms by A.Tanenbaum and Maarten van Steen, Prentice Hall of India.
3. Advanced Concepts in Operating Systems by M. Singhal and N. Shivaratri,Tata McGraw Hill.

References Books

1. Distributed Systems - An Algorithmic Approach by Sukumar Ghosh, Chapman and Hall/CRC.
2. Distributed Algorithms: Principles, Algorithms, and Systems by D. Kshemkalyani and M. Singhal, Cambridge University Press.
3. Distributed Systems - An Algorithmic Approach by Sukumar Ghosh, Chapman and Hall/CRC.
4. Distributed Algorithms: Principles, Algorithms, and Systems by D. Kshemkalyani and M. Singhal, Cambridge University Press.

Course Name: **Machine Learning**
Course Code: **CS-382**
Course Type: **Stream Core-I**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To impart knowledge about the concepts of machine learning.
- To introduce the different types machine learning algorithms and their applications.
- To enable the students to understand the mathematics behind learning algorithms and construct novel solutions to real world problems.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Machine Learning:** Problems, data, and tools, Visualization tools, Decision Tree Learning, Artificial Neural Networks, Bayesian Learning, Deep Learning, Instance-Based Learning.	**6 L**
UNIT-02	**Regression Techniques:** Linear regression, Logistic regression, online gradient descent, Kernel Methods, SSE, closed form, normal equations, features, Overfitting and complexity, training, validation, test data, Classification problems, decision boundaries, nearest neighbor methods.	**6L**
UNIT-03	**Probability and classification:** Bayes optimal decisions, Naive Bayes and Gaussian class-conditional distribution, Linear classifiers: Bayes Rule and Naive Bayes Model. Radial Basis Function Networks, Support Vector Machines, Genetic Algorithms, Reinforcement Learning.	**6L**
UNIT-04	**Ensemble methods:** Bagging, random forests, boosting, Unsupervised learning: clustering, k-means, hierarchical agglomeration, Latent space methods, PCA, Text representations, naive Bayes and multinomial models. Clustering and latent space models, VC-dimension, structural risk minimization, margin methods and support vector machines (SVM), Machine Learning Applications.	**6 L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Develop an understanding what is involved in learning models from data.
CO2: Understand a wide variety of learning algorithms.
CO3: Apply principles and algorithms to evaluate models generated from data.
CO4: Apply the algorithms to a real-world problem

Text Books:

1. Introduction to Machine Learning by Ethem Alpaydin, PHI Learning.
2. Machine Learning: An Algorithmic Perspective by Stephen Marsland, Chapman and Hall/CRC.
3. Pattern Recognition and Machine Learning by Christopher M. Bishop, Springer.

Reference Books:

4. Machine Learning by Tom Mitchell, McGraw Hill Education.
5. Fundamentals of Machine Learning for Predictive Data Analytics, John D. Kelleher, Brian Mac Namee, and Aoife D'Arcy, MIT Press.

Course Name: **Engineering Economics and Accountancy**
Course Code: **HS-321**
Course Type: **Institute Core**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To impart knowledge about Economics and its applicability to the Engineers
- To introduce the fundamental concepts of economics
- To enable the students to understand the factors that cause the changes in economic conditions of the entrepreneur.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Engineering Economics:** Definitions, Nature, Scope and application; Difference between Micro Economics and Macro Economics; Theory of Demand & Supply: Meaning, Determinants, Law of Demand, Elasticity of demand, Demand Forecasting, Law of Supply, Equilibrium between Demand & Supply.	**6L**
UNIT-02	**Production and Cost**: Production functions, Least Cost combination, Laws of Returns to Scale. Cost and Cost curves, Revenue and Revenue curves, Break even analysis.	**6L**
UNIT-03	**Costing and Appraisal:**Cost elements, Economic cost, Accounting cost, Standard cost, Actual cost, Overhead cost, Cost control, Criteria of project appraisal, Social cost benefit analysis	**3L**
UNIT-04	**Money**: Meaning, Functions, Types. **Banking:** Meaning, Types, Functions, Central Bank: its Functions, concepts CRR, Bank Rate, Repo Rate, Reverse Repo Rate, SLR.	**3L**
UNIT-05	**Depreciation:** Meaning of depreciation, causes, object of providing depreciation, factors affecting depreciation, Methods of Depreciation: Straight line method, Diminishing balance method, Annuity method and Sinking Fund method	**4L**
UNIT-06	**Financial Accounting:** Double entry system (concept only), Rules of Double entry system, Journal(Sub-division of Journal) , Ledger, Trial Balance Preparation of final accounts-Trading Account. Profit and Loss account, Balance Sheet.	**4L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Familiarize with the concepts of Engineering economics i.e. economic theory, decision making and management.

CO2: Understand and apply the fundamentals of microeconomics in achieving the consumers and entrepreneurs/manufacturers motive of maximize satisfaction & maximize profit respectively by using by optimization techniques.

CO3: Learn about the various concepts of cost and their role in determining the producer's behavior.

CO4: Money, Banking helps in increasing the trade in the economy and telling how it is going to affect the cost and profitability of the entrepreneur

CO5: Understand the concept of depreciation, and valuation.

CO6: The Trading , Profit and loss account and the Balance sheet that a manufacturer needs to submit to the government and to attract the investors for making the investment in their company by purchasing the shares and debentures issued by them.

Books and References

1. Principles of Micro Economics by Mceachern & Kaur, Cengage Publication.
2. Managerial Economics: by Craig Peterson & W Cris Lewis, PHI Publication.
3. Modern Microeconomics: by A. Koutsoyiannis, Macmillan.
4. Managerial Economics Theory and Applications: by D. M. Mithani. Himalaya Publication House.
5. Fundamental of Managerial Economics: Mark Hirschey, South Western Educational Publishing.
6. Engineering Economics: by Degramo, Prentice Hall.
7. Financial Accounting – A Managerial Perspective by R. Narayanaswamy, PHI.
8. Introduction to Accounting by J.R. Edwards, Marriot, Sage Publication.
9. Cost Accounting by Jawahar Lal, Tata McGraw Hill.
10. Project planning Analysis, Selection, Implementation and Review:by Prasanna Chandra,Tata McGraw Hill

Course Name:	**Digital Image Processing Lab**
Course Code:	**CS-324**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To perform image sampling and quantization.
- To analyze spatial and intensity resolution of images.
- To perform intensity transformation of images.
- To apply Discrete Fourier Transform on image and study its properties.
- To study the histogram and histogram equalization.

List of Experiments

1. Image sampling and quantization.
2. Analysis of spatial and intensity resolution of images.
3. Intensity transformation of images and DFT analysis of images.
4. Transforms (Walsh, Hadamard, DCT, Haar).
5. Histogram Processing and Basic Thresholding functions.
6. Image Enhancement-Spatial filtering and Image Enhancement- Filtering in frequency domain.
7. Image segmentation – Edge detection, line detection and point detection.
8. Basic Morphological operations.
9. Region based Segmentation.
10. Segmentation using watershed transformation.
11. Analysis of images with different color models.
12. Study of DICOM standards.
13. Image compression techniques and Image restoration.

Note: The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand basic concepts and develop application of image processing using MATLAB tools.
CO2: Perform analysis of images with different color models.
CO3: Perform region based segmentation of image.

Course Name:	**Database Management Systems Lab**
Course Code:	**CS-325**
Course Type:	**Discipline Core**
Contact Hours/Week: **2P**	Course Credits: **01**

Course Objectives

- To impart knowledge about the DBMS concept, data models and architecture.
- To introduce the fundamental concepts relevant to ER model and its mapping to relational model, relational algebra and SQL.
- To enable the students to understand the factors that cause the concurrency and recovery strategies for DBMS.

List of Experiments

Gather the Required Information, Draw ER Diagrams, Map ER Diagrams to Tables, Perform Normalization, Write Procedures, Execute Queries, Create User Interfaces and Generate Reports:

1. Online Examination System/Development of Online Course Portal for the Campus.
2. Student Attendance Management System
3. Student Information System
4. Student Result Alert System
5. Time Table Management System
6. Examination Date Sheet Management System
7. Supplementary Examination Registration System
8. Training and Placement Cell Management
9. Hostel Allotment System/Campus Network Complaint System
10. Hospital Management System/Civil/ Electrical Works Repair Information System

Note: The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand basic concepts and develop application using DBMS tools and techniques.

CO2: Use relational data model, entity-relationship model, relational database design, relational algebra and SQL in practical applications

CO3: Improve the database design by normalization.

Course Name: **Advance Computer Architecture**
Course Code: **CS-411**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about examining the qualitative and quantitative computer design tradeoffs.
- To introduce the fundamental concepts relevant to art of selecting and interconnecting hardware components to create a computer that meets functional, performance and cost goals.
- To enable the students to understand the basic non-classical architectures such as parallel processors, multi-core chips, pipelined and VLIW machines.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Review of basic computer architecture, quantitative techniques in computer design, measuring and reporting performance, CISC and RISC processors.	**6 L**
UNIT-02	**Pipelining:** Basic concepts, instruction and arithmetic pipeline, data hazards, control hazards, and structural hazards, techniques for handling hazards, Exception handling, Pipeline optimization techniques. Compiler techniques for improving performance.	**8 L**
UNIT-03	**Hierarchical Memory Technology:** Inclusion, Coherence and locality properties; Cache memory organizations, Techniques for reducing cache misses; Virtual memory organization, mapping and management techniques, memory replacement policies.	**8 L**
UNIT-04	**Instruction-Level Parallelism:** basic concepts, techniques for increasing ILP, superscalar, super pipelined and VLIW processor architectures. Array and vector processors. Multiprocessor architecture: taxonomy of parallel architectures.	**4L**
UNIT-05	**Centralized Shared-Memory Architecture:** Synchronization, memory consistency, interconnection networks.	**4 L**
UNIT-06	**Distributed Shared-Memory Architecture:** Cluster computers. Non von Neumann architectures: data flow computers, reduction computer architectures, systolic architectures.	**6 L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand that how hardware and software (especially the operating system and compilers) must work synergistically together to provide optimum throughput.

CO2: Understand the elements of modern computer along with measures of performance.

CO3: Understand the concepts of pipelining, multiprocessors, parallel processors, etc.

Text Books:

1. Computer Organization and Design: A Hardware/Software Interface by David Patterson and John Henessey, Morgan Kaufmann publication.
2. Computer Architecture: A Quantitative Approach by John Henessey & David Patterson, Morgan Kaufmann
3. Structured Computer Organization by Andrew S. Tanenbaum and Todd Austin, Prentice Hall of India.

References Text:

1. Advanced Computer Architecture: Parallelism, Scalability, Programmability by Kai Hwang, Tata McGraw Hill
2. Using MPI : portable parallel programming with the message-passing interface, William Gropp, Ewing Lusk, Anthony Skjellum, Cambridge MIT Press.
3. Introduction to Parallel Computing, A Grama, A Gupta, G Karypis, and V Kumar, Addison-Wesley, 2003.

Course Name: **Information Security & Privacy**
Course Code: **CS-412**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the network security to predict and classify attacks on a system/network
- To introduce the fundamental concepts relevant to Cryptographic Techniques.
- To enable the students to understand security threats to ICT infrastructure using modern tools such as firewalls, UTMs, etc.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Standards Organizations, Security Components OSI Security Architecture, Aspects of Security, Passive Attacks, Active Attacks, Security Services (X.800), Model for Network Security, Model for Network Access Security, Symmetric Cipher Model, Cryptography Classification, Cryptanalysis, Substitution: Other forms, Poly-alphabetic Substitution Ciphers, One-Time Pad, Transposition (Permutation) Ciphers, Product Ciphers.	7L
UNIT-02	**Number Theory and Prime Numbers:** Groups, Rings, and Fields, Modular Arithmetic, Euclid"s Algorithm, Finite Fields of the Form GF(p), Polynomial Arithmetic, Finite Fields of the Form GF(2^n). Generation of large prime numbers, Prime factorization, Euler Totient Function ø(n), Euler's Theorem, Primality Test- Fermat's Little Theorem, Baillie-PSW, Solovay-Strassen, Miller Rabin Algorithm, AKS Algorithm, Cyclotomic primality test, Elliptic Curve Primality Test, Prime Distribution, Chinese Remainder Theorem, Primitive Roots, Discrete Logarithms	7L
UNIT-03	**Cryptographic Techniques:** Perfect security, Feistel Cipher Structure, Block Cipher- DES, differential and Linear Cryptanalysis, Avalanche Effect, Double-DES, Triple-DES, Cipher modes of operations: block and stream mode, AES, International Data Encryption Algorithm (IDEA), Blowfish Algorithm; RC4; Pseudo number generation, Blum Blum Shub Generator.	6L
UNIT-04	**Public-Key Cryptography and Message Authentication:** The Key Distribution Problem, Public-Key Cryptosystems, The RSA Algorithm, The Key Management riddle, The Diffie-Hellman Key Exchange, Elliptic Curve Cryptography, Message Authentication, requirements and functions, Message Authentication Codes, Hash Functions, Birthday Problem, SHA-X, SHA-512, Authentication, Access control policies, The Message Digest (MD5) Algorithm, HMAC fundamentals, Digital Signature basics, Authentication Protocols, The Digital Signature Standard, Kerberos Authentication scheme, The X.509 Directory Authentication scheme.	6L
UNIT-05	**Security Protocols:** Secure User Authentication, Mail security, PGP, database security, File system security, Program security, Memory security, Session security, SSH, Web security, Replay Attacks, Needham Schroeder Protocol, IPSec, SSL, IEEE 802.11, Wired Equivalent Privacy (WEP).	6L
UNIT-06	**Intrusion detection:** Intrusion vs. Extrusion Detection, Examples of Intrusion, Categories of Intruders, Hacker Behavior Example, Criminal Enterprise Behavior, Insider Behavior Example, Intrusion Techniques, Password Guessing and Capture, Notification Alarms, Types of IDS, Sample Signatures, Anomaly Based IDS, Statistical Anomaly Detection, Audit Records, Rule-Based Intrusion Detection, Types of ID, Host vs. Network IDS, Honeypots.	6L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Analyze the basic concepts of network security to predict and classify attacks on a system/network
CO2: Illustrate the process of cryptographic algorithms for information hiding
CO3: Understand and apply authentication techniques to provide secure communication
CO4: Assess the security threats to ICT infrastructure using modern tools such as firewalls, UTMs, etc.

Text Books:

1. Cryptography and Network Security: Principles and Practice by William Stallings, Pearson Education.
2. Cryptography: Theory and Practice by D Stinson, Chapman & Hall.
3. Network Security by C. Kaufman, R. Perlman and M. Spenser, Prentice Hall of India.

Reference Books:

1. Internet Security and Firewalls by S. Bellovin and W. Chesvick, Addison-Wesley, Reading.
2. Introduction to Cryptography with Coding Theory by Trappe & Washington, Prentice-Hall.
3. Cryptography & Network Security by Behrouz A. Forouzan, McGraw Hill Education.

Course Name: **Data Warehousing & Data Mining**
Course Code: **CS-413**
Course Type: **Discipline Core**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about fundamental Data warehousing theory and its architecture.
- To introduce the fundamental concepts demonstrating the competency in the use of CRISP-DM, the Cross-Industry Standard Process for Data Mining.
- To enable the students to understand current data mining techniques and applications, such as text mining, mining genomics data, and other current issues.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Data warehousing Definition, usage and trends. DBMS vs Datawarehouse, Data marts, Metadata, Multidimensional data mode, Data cubes, Schemas for Multidimensional Database: stars, snowflakes and fact constellations.	4L
UNIT-02	**Architecture:** Datawarehouse process & architecture, OLTP vs OLAP, ROLAP vs MOLAP, types of OLAP, servers, 3-Tier Datawarehouse architecture, distributed and virtual Data-warehouses, Datawarehouse manager.	5L
UNIT-03	**Implementation:** Data warehouse implementation, computation of data cubes, modeling OLAP data, OLAP queries manager, Datawarehouse back end tools, complex aggregation at multiple granularities, tuning and testing of Datawarehouse.	6L
UNIT-04	**Data Mining, Association rules and Classification:** Types of Data, Data Mining Functionalities, Interestingness of Patterns, Classification of Data Mining Systems, Data Mining Task Primitives, Integration of a Data Mining System with a Datawarehouse.	10L
UNIT-05	**Issues with Data Preprocessing:** Mining Frequent Patterns, Associations and Correlations, Constraint Based Association Mining, Classification and Prediction: Decision Tree Induction, Bayesian Classification, Rule Based Classification, Classification by Back propagation, Support Vector Machines, Associative Classification, Lazy Learners, Prediction.	5L
UNIT-06	**Cluster Analysis:** Cluster Analysis, Types of Data, Categorization of Major Clustering Methods, K-means, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid Based Methods, Model-Based Clustering Methods, Clustering High Dimensional Data, Constraint Based Cluster Analysis, outlier Analysis. Other Trends in Data Mining: Decision tree knowledge discovery through Neural Networks & Genetic Algorithm, Rough Sets and Fuzzy techniques	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Describe fundamental data warehousing theory and its architecture.
CO2: Understand data mining concept and functionalities.
CO3: Understand and apply a wide range of clustering, association, and classification algorithms.
CO4: Understand and apply the most current data mining techniques for real-life applications, such as text mining, mining genomics data, and other current issues.

Text Books:

1. Data Warehousing in the Real World by Sam Anahory and Dennis Murray, Pearson.
2. Data Mining-Concepts & Techniques by Jiawei Han and Micheline Kamber, Morgan Kaufmann.
3. Data Mining by Pieter Adriaans and Dolf Zantinge, Pearson.

Reference Books:

1. Data Warehousing, Data Mining and OLTP by Alex Berson, McGraw Hill.
2. Developing the Data Warehouses by W. H Ionhman C. Klelly, John Wiley & Sons.
3. Managing the Data Warehouses by W. H. Inman and C.L. Gassey, John Wiley & Sons.

Course Name: **Information Theory and Coding**
Course Code: **CS-431**
Course Type: **Discipline Elective V**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To learn concepts in information theory, and the performance characteristics of an ideal communications system.
- To impart knowledge about the fundamentals of Information coding and its applications.

Unit Number	Course Content	Contact Hours
UNIT-01	**Basics of Information Theory:** Noise and its sources, Introduction to Information Theory, Information and entropy, properties of entropy of a binary memoryless source, Measure of Information, Shannon's Theorem.	**8L**
UNIT-02	**Source codes:** Source Coding, Coding parameters, run length coding, arithmetic coding, Shannon Fano coding, Huffman coding, Lempel Ziv coding, LZW coding, channel coding, Channel capacity, noisy channel, audio and video coding standards.	**8L**
UNIT-03	**Coding Theory:** Types of codes, error control strategies, linear block codes, generator matrices, parity check matrices, encoder syndrome and error detection minimum distance, error correction and error detection capabilities.	**8L**
UNIT-04	**Cyclic and BCH Codes:** Generation, Syndrome computation and error correction, Decoding, Cyclic Hamming codes, Cyclic redundancy check. BCH Code generation, decoding, Non-binary BCH codes, weight distribution.	**8L**
UNIT-05	**Convolution Codes:** Tree and trellis codes, encoding, properties, decoding, construction, implementation, modification, Turbo coding and decoding, Interleaving techniques.	**5L**
UNIT-06	**Introduction to Cryptography:** Plain text, cipher text, substitution and transposition, encryption, decryption, symmetric and asymmetric cryptography, steganography, hashing, key exchanges during encrypted communication.	**5L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand that how the quantity of information could be measure.
CO2: Understand the concept and properties of entropy and mutual information as applied to information.
CO3: Construct compact and non-compact codes for a given data ensemble.
CO4: Understand the basic concepts of encryption and decryption.

Text Books:

1. Information Theory Coding and Cryptography by R. Bose, Tata McGraw Hill.
2. The Theory of Error Correcting Codes by F. J. MacWilliams and N. J. A. Sloane, Elsevier.
3. Coding and Information Theory by S. Roman, Springer.
4. Introduction to Data Compression by Khalid Sayood.

Reference Books:

1. The Theory of Information and Coding by R. J. McEliece, Cambridge University Press.
2. Elements of Information Theory by T. M. Cover and J. A. Thomas, Wiley.

Course Name: **Soft Computing**
Course Code: **CS-432**
Course Type: **Discipline Elective-V**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the basic principles, techniques, and applications of soft computing.
- Provide the mathematical background for carrying out the optimization associated with neural network learning.
- Develop the skills to gain basic understanding of the areas of Soft Computing including Artificial Neural Networks, Fuzzy Logic and Genetic Algorithms.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Introduction to Neuro fuzzy and Soft Computing, hard computing, Need for soft computing; Neurons and neural networks, Fuzzy set theory, Fuzzy Rules, Fuzzy Reasoning, Fuzzy inference System.	5 L
UNIT-02	**Fuzzy Logic:** Introduction to Fuzzy logic, Fuzzy sets and membership functions, Operations on Fuzzy sets, Fuzzy relations, rules, propositions, implications and inferences, Defuzzification techniques, Fuzzy logic controller design, Some applications of Fuzzy logic.	7 L
UNIT-03	**NeuroFuzzy Modeling:** Genetic Algorithms, Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques, Basic GA framework and different GA architectures, GA operators: Encoding, Crossover, Selection, Mutation, etc. Solving single-objective optimization problems using GAs. Applications of Soft Computing to Signal Processing, Image Processing, Forecasting, XOR Problem traveling salesman problem.	8L
UNIT-04	**Multi-Objective Optimization Problem Solving:** Concept of multi-objective optimization problems (MOOPs) and issues of solving them, Multi-Objective Evolutionary Algorithm (MOEA), Non-Pareto approaches to solve MOOPs, Pareto-based approaches to solve MOOPs, Some applications with MOEAs.	8 L
UNIT-05	**Neural Networks:** Radial basis and recurrent neural networks, Hopfield Networks, Comparison of RBF and MLP Network, Running Algorithms	5 L
UNIT-06	**Combined techniques:** Genetic Algorithms–Fuzzy Logic, Genetic Algorithms–Neural Networks, Neural Networks– Fuzzy Logic.	6L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Comprehend the fuzzy logic and the concept of fuzziness involved in various systems and fuzzy set theory

CO2: Understand the concepts of fuzzy sets, knowledge representation using fuzzy rules, approximate reasoning, fuzzy inference systems, and fuzzy logic.

CO3: Understand the fundamental theory and concepts of neural networks, Identify different neural network architectures, algorithms, applications and their limitations.

CO4: Understand appropriate learning rules for each of the architectures and learn several neural network paradigms and its applications.

Text Books:

1. Principles of Soft Computing by S N Sivanandam, S N Deepa, Wiley India.
2. Soft Computing and Intelligent System Design by Fakhreddine O Karray, Clarence D Silva. Pearson Edition,
3. Artificial Intelligence and Soft Computing - Behavioural and Cognitive Modeling of the Human Brain by Amit Konar, CRC press, Taylor and Francis Group.

Reference Books:

1. Artificial Intelligence by Elaine Rich and Kevin Knight, TMH.
2. Artificial Intelligence by Patric Henry Winston – Third Edition, Pearson Education.
3. A first course in Fuzzy Logic by Hung T Nguyen and Elbert A Walker, CRC. Press Taylor and

Course Name: **Big Data Analytics**
Course Code: **CS-433**
Course Type: **Discipline Elective-V**

Contact Hours/Week: 3L Course Credits: **03**

Course Objectives

- To understand the Big Data Platform and its Use cases.
- Apply analytics on Structured and Unstructured Data.
- Acquire the knowledge and working on Big Data platforms

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Big Data:** Types of Digital Data, Introduction to Big Data, Big Data Analytics, Relational Databases & SQL, Data Cleansing and Preparation,	**4 L**
UNIT-02	**Hadoop:** History of Hadoop, Apache Hadoop, Analysing Data with Unix tools, Analyzing Data with Hadoop, Hadoop Streaming, IBM Big Data Strategy, Infosphere Big Insights and Big Sheets.	**5 L**
UNIT-03	**HDFS (Hadoop Distributed File System):** The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures.	**5 L**
UNIT-04	**Map Reduce:** Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features. Hadoop Eco System Pig: Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators.	**10 L**
UNIT-05	**Hive:** Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions. Hbase: HBasics, Concepts, Clients, Example, Hbase Versus RDBMS.	**6 L**
UNIT-06	**Advanced Topics:** Big SQL, Data Analytics with R, Machine Learning: Introduction, Supervised Learning, Unsupervised Learning, collaborative filtering. Big Data Analytics with Big R.	**6 L**

Course Outcomes

CO1: Describe and analyze various Big Data platforms.
CO2: Develop Big Data Solutions using Hadoop Eco System.
CO3: Apply Machine Learning Techniques using R

Text Books:

1. Data Science for Business by F. Provost and T. Fawcett, O'Reilly Media.
2. Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics by Bill Franks, John Wiley & Sons.
3. Hadoop: The Definitive Guide by Tom White, O'reilly Media.

Reference Books:

1. Big Data and Business Analytics by Jay Liebowitz, Auerbach Publications, CRC Press.
2. Big Data: Concepts, Technology and Architecture, Balamarugan Balusamy, Nandhini Abirami R, Seifedine Kadry and Amir Gandomi, Wiley.
3. Big Data in Practice: How 45 Successful Companies Used Big Data Analytics to Deliver Extraordinary Results, Bernard Marr, Wiley.

Course Name: **Advance Mobile Communication**
Course Code: **CS-451**
Course Type: **Stream Core-II**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To understand the basic cellular system concepts.
- To have an insight into the interference, frequency management and handoff management in cellular mobile system.
- To go in depth for understanding the popular GSM cellular mobile standard and wireless standards.

Unit Number	Course Content	Contact Hours
UNIT-01	**Evolution from 1G to 5G:** Analog voice systems in 1G; digital radio systems in 2G, voice and messaging services, TDMA based GSM, CDMA, 2.5G (GPRS), 2.75G (EDGE). IMT2000: 3G UMTS, W-CDMA, HSPA, HSPA+, 3G services and data rates; IMT Advanced: 4G, LTE, VOLTE, OFDM, MIMO, LTE Advanced Pro (3GPP Release 13+); IMT2020: 5G, enhancements in comparison to IMT Advanced.	**7L**
UNIT-02	**Basics of 5G:** 5G potential and applications, Usage scenarios: enhanced mobile broadband (eMBB), ultra-reliable low latency communications (URLLC), massive machine type communications (MMTC), D2D communications, V2X communications.	**7L**
UNIT-03	**Spectrum for 5G:** spectrum access/sharing: millimeter Wave communication, channels and signals/waveforms in 5G, carrier aggregation, small cells, dual connectivity.	**5L**
UNIT-04	**5G Network:** New Radio (NR), Standalone and non-standalone mode; non-orthogonal multiple access (NOMA); massive MIMO, beam formation, FAPI: PHY API Specification, flexible frame structure, Service Data Adaptation Protocol (SDAP), Centralized RAN, open RAN; multi-access edge computing (MEC); software defined networking (SDN), network function virtualization (NFV); network slicing; restful API for service-based interface; private networks.	**7L**

Course Outcomes

Upon successful completion of the course, the students will be able to
CO1: Discuss cellular radio concepts.
CO2: To have knowledge of the mobile system specifications.
CO3: Classify frequency and handoff management techniques in mobile communication.
CO4: Outline cellular mobile communication standards.
CO5: Analyze various methodologies to improve the cellular capacity.

Text Books:

1. Mobile Cellular Telecommunications: Analog and Digital Systems by W. C. Y. Lee; Tata McGraw Hill Publication.
2. Wi-Fi, Bluetooth, Zigbee and WiMax by H. Labiod, H. Afifi and C. D. Santis, Springer.
3. Wireless Communications: Principles and Practice by T. S. Rappaport; Pearson Publication.

Reference Books:

1. Wireless Communications and Networks: 3G and Beyond by I. S. Misra; Tata McGraw Hill Publication.
2. Wireless and Digital Communications by K. Feher; PHI Publication.
3. 4G, LTE-Advanced Pro and The Road to SG by Erik Dahlman.

Course Name: **Deep Learning**
Course Code: **CS-452**
Course Type: **Stream Core-II**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To introduce the students to the foundational concepts of deep learning.
- To familiarize students with the different aspects of training deep neural networks.
- To enable the students to build deep neural networks for various real-world applications.

Unit Number	Course Content	Contact Hours
UNIT-01	**Basics:** Review of Linear Algebra, Review of Probability and Information Theory, Review of Numerical Computations, Review of Machine Learning, MP Neurons, Perceptrons, Introduction to Deep Learning, Deep Feed Forward Neural Networks	**5L**
UNIT-02	**Training Deep Networks:** Backpropagation, Gradient Descent, Optimization for training Deep neural networks, GD, Momentum based GD, Nestrov Accelerated GD, AdaGrad, RMS Prop, Tricks for Improving the Learning, Better weight initialization, Regularization, Dropout, Batch Normalization	**7L**
UNIT-03	**Deep Learning Architectures:** Convolutional Neural Networks (CNNs), Visualizing CNNs, Recurrent neural networks (RNN) and long-short term memory (LSTM), Introduction to DL packages/Important architectures.	**7L**
UNIT-04	**Advanced Methods & Applications:** Encoder-Decoder architectures, Attention mechanism, Applications: action recognition, sentiment classification.	**5L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Learn the fundamental principles of deep learning for computer vision.
CO2: Identify the deep learning algorithms for various types of learning tasks in vision domain.
CO3: Implement deep learning algorithms and solve real-world problems.

Books:

1. Deep Learning by Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT Press, 2016.
2. Deep Learning with Python by François Chollet, Manning Publisher

References:

1. Pattern Recognition and Machine Learning by Christopher M. Bishop, Springer.
2. Bayesian Reasoning and Machine Learning by David Barber, Cambridge University Press.
3. Mathematics for Machine Learning by Marc Peter, Aldo Faisal, Cheng Soon, Cambridge University Press.

Course Name: **Internet of Things**
Course Code: **CS-471**
Course Type: **Stream Core III**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To impart knowledge about Internet of Things.
- To introduce the fundamental concepts relevant to design issues related to Internet of Things.
- To enable the students to understand the basic principles of communication in Internet of Things

Unit Number	Course Content	Contact Hours
UNIT-01	**IoT Introduction and Fundamentals:** Introduction to loT, deployment Benefits/Challenges of deploying an IoT, IoT components: Digital Signal Processing, Data transmission, Choice of channel (wired/wireless), back-end data analysis. Understanding packaging and power constraints for loT implementation.	**7L**
UNIT-02	**Signals, Sensors, Actuators, Interfaces:** Introduction to sensors & transducers, Introduction to electrodes & biosensors, Static and dynamic characteristics of sensors, Different types of sensors, Selection criteria's for sensors/transducers, Signal conditioning modules of loT system, Energy and power considerations, Introduction to actuators, Different types of actuators, Interfacing challenges, Modules of data acquisition system, Wireless sensor node structure, positioning topologies for loT infrastructure.	**7L**
UNIT-03	**Communication and Networking in loT:** Review of Communication Networks, Challenges in Networking of lot Nodes, range, bandwidth Machine-to-Machine (M2M) and IoT Technology Fundamentals, Medium Access Control (MAC) Protocols for M2M Communications Standards for the loT Basics of 5G Cellular Networks and 5G IoT Communications, Low-Power Wide Area networks (LPWAN)Wireless communication for loT: channel models, power budgets, data rates. Networking and communication aspects: IPv6, 6LoWPAN, COAP, MQTT, LORAWAN, Zigbee Operating Systems need and requirements for IoT.	**7L**
UNIT-04	**loT Security:** IoT Data Analytics. Cryptographic algorithms, Analysis of Light weight cryptographic algorithms, homomorphic encryption, Key exchange using Elliptical Curve Cryptography. Comparative analysis of Cryptographic Library for loT.	**5L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Illustrate the fundamentals of loT.
CO2: Identify suitable hardware and interfaces for loT deployments.
CO3: Understand basic communication protocols for IOT.
CO4: Illustrate data analytics and security for loT.

Text Books:

1. Internet of Things Principles and Paradigms by Rajkumar Buyya and Amir Vahid Dastjerdi, Elsevier.
2. From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence by Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stamatis Karnouskos, Stefan Avesand and David Boyle, Academic Press.
3. Sensors, Actuators and Their Interfaces by N. Ida, Scitech Publishers.

Reference Books:

1. Arshdeep Bahga and Vijay Madisetti, "Internet of Things, a Hands-on approach" Universities Press (India) Pvt. Ltd. 2017.
2. Rajkumar Buyya, Amir Vahid Dastjerdi, "Internet of Things Principles and Paradigms"Copyright © 2016 Elsevier Inc.
3. William Stallings," Foundations of Modem Networking: SDN, NFV, QoE, IoT, and Cloud" Publisher: Addison-Wesley 2015

Course Name: **Pattern Recognition**
Course Code: **CS-472**
Course Type: **Stream Core III**

Contact Hours/Week: **2L** Course Credits: **02**

Course Objectives

- To provide comprehensive knowledge of fundamental concepts of Pattern Recognition.
- To develop the mathematical tools required for the pattern recognition.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Fundamental concepts and blocks of a typical pattern recognition system. Decision functions- role and types, pattern and weight space, properties and implementation of decision functions.	5L
UNIT-02	**Features:** Feature identification, selection and extraction. Distance measures, clustering transformation and feature ordering, clustering in feature selection, feature selection through maximization and approximations.	6L
UNIT-03	**Methods:** Bayesian decision theory, supervised learning from data, parametric and non-parametric estimation of density functions, Bayes and nearest neighbor classifiers, Pattern classification by distance functions.	6L
UNIT-04	**Template Matching:** Introduction, Measures Based on Optimal Path Searching Techniques, Measures Based on Correlations, Deformable Template Models, Content Based Information Retrieval: Relevance Feedback, Applications: Statistical and neural network-based pattern classifiers in speech recognition/ image recognition.	7L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand basic concepts of Machine Learning.
CO2: Categorize the various pattern recognition techniques into supervised and unsupervised.
CO3: Discuss the applications of pattern recognition in various applications

Text Books:

1. J.I. Tou & R.C. Gonzalez, Pattern Recognition Priciples, Addition-Wesley.
2. R. Schalkoff, Pattern Recognition - Statistiucal, Structural and Neural Approaches, John Wiley, 1992.
3. P.A. Devijer & J. Kittler, Pattern Recognition - A Statistical Approach, Prentice-Hall.

Reference Books:

1. Christopher. M. Bishop, 'Pattern recognition and machine learning, Springer, 2006.
2. Human Activity Recognition Using Wearable Sensors and Smartphones, Miguel A. Labrador Oscar D. Lara Yejas, CRC Press.
3. Pattern Recognition and Machine Learning, Christopher M. Bishop, Springer New York, NY.

Course Name:	**Information Security and Privacy Lab**
Course Code:	**CS-414**
Course Type:	**Discipline Core**
Contact Hours/Week: 2P	Course Credits: **01**

Course Objectives

- To configure virtual networks using network simulator.
- To install and exploit security tools for protecting a network.
- To implement cryptographic algorithm for building a secure communication network.
- To exploit the vulnerabilities in a LAN environment and launch attacks.
- To analyze the network packet using Wireshark.
- To perform the web penetration testing using Burp suite.
- To perform vulnerability assessment of wireless devices.
- To exploit vulnerabilities in the systems.
- To perform the log analysis using Splunk.
- To find vulnerable apps in play store and perform static and dynamic analysis on it.

List of Experiments

1. LAN based Network Security
2. Network reconnaissance and Protection.
3. Application of Cryptographic algorithms using Crypto tools
4. LAN based insider attacks
5. Network Packet analysis using Wireshark.
6. Web Penetration testing using Burp Suite.
7. Wireless Security Lab
8. Exploiting the vulnerabilities on a system
9. Log analysis using Splunk
10. Mobile & Smart phone security Lab

Note: *The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.*

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Implementation of various network exploits and its mitigation techniques using simulators and real devices.
CO2: To exploit vulnerabilities in LAN, wireless devices and identify the same using penetration testing
CO3: Exploring the reverse engineering techniques for proper classification of Benign and malicious Desktop/ Android applications
CO4: Implementation of Intrusion detection system by applying machine learning algorithms

Course Name:	**Data Warehousing & Data Mining Lab**
Course Code:	**CS-415**
Course Type:	**Discipline Core**
Contact Hours/Week: 2P	Course Credits: **01**

Course Objectives

- Learn how to build a data warehouse and query it (using open source tools like Pentaho Data Integration Tool, Pentaho Business Analytics).
- Learn to perform data mining tasks using a data mining toolkit (such as open source WEKA).
- Understand the data sets and data preprocessing.
- Demonstrate the working of algorithms for data mining tasks such association rule mining, classification, clustering and regression.
- Exercise the data mining techniques with varied input values for different parameters.
- To obtain Practical Experience Working with all real data sets.

List of Experiments

1. Build Data Warehouse and Explore WEKA
2. Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets.
3. Demonstrate performing classification on data sets
4. Demonstrate performing clustering on data sets
5. Demonstrate performing Regression on data sets.
6. Task 1: Credit Risk Assessment. Sample Programs using German Credit Data.
7. Task 2: Sample Programs using Hospital Management System.
8. Beyond the Syllabus -Simple Project on Data Preprocessing

Note: *The concerned Course Coordinator will prepare the actual list of experiments/problems at the start of semester based on above generic list.*

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Able to understand the various kinds of tools.
CO2: Demonstrate the classification, clustering and etc. in large data sets.
CO3: Able to add mining algorithms as a component to the exiting tools.
CO4: Ability to apply mining techniques for realistic data.

Course Name: **Reinforcement Learning**
Course Code: **CS-461**
Course Type: **Stream Elective-I**

Contact Hours/Week: 3L Course Credits: **03**

Course Objectives

- To impart knowledge about the key features of reinforcement learning that distinguishes it from AI and non-interactive machine learning.
- To give an application problem (e.g. from computer vision, robotics, etc), decide if it should be formulated as a RL problem; if yes be able to define it formally (in terms of the state space, action space, dynamics and reward model), state what algorithm (from class) is best suited for addressing it.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Reinforcement Learning, Examples and Elements of Reinforcement Learning Limitations and Scope, An Extended Example: Tic-Tac-Toe, Early History of Reinforcement Learning.	**4 L**
UNIT-02	**Finite Markov Decision Processes:** The Agent–Environment Interface, Goals and Rewards, Returns and Episodes, Policies and Value Functions, Optimal Policies and Optimal Value Functions, Optimality and Approximation, Dynamic Programming: Policy Evaluation, Policy Improvement, Policy Iteration, Value Iteration, Asynchronous Dynamic Programming, Generalized Policy Iteration, Efficiency of Dynamic Programming	**8 L**
UNIT-03	**Monte Carlo Prediction:** Monte Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off policy Prediction via Importance Sampling, Incremental Implementation, Off policy Monte Carlo Control	**6 L**
UNIT-04	**Temporal-Difference Learning:** Sarsa: On-policy TD Control, Q-learning: Off policy TD Control, Maximization Bias and Double Learning, Games, After states, and Other Special Cases.	**6 L**
UNIT-05	**n-step Bootstrapping:** n-step TD Prediction, n-step Sarsa, n-step Off policy Learning, *Per-decision Methods with Control Variates, Off-policy Learning Without Importance Sampling: The n-step Tree Backup Algorithm.	**6 L**
UNIT-06	**Planning and Learning:** Planning and Learning with Tabular Methods, Approximate Solution Methods, On-policy Prediction with Approximation, On-policy Control with Approximation, *Off policy Methods with Approximation, Eligibility Traces.	**6 L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Define the key features of reinforcement learning that distinguishes it from AI and non-interactive machine learning

CO2: Implement in code common RL algorithms and describe (list and define) multiple criteria for analyzing RL algorithms and evaluate algorithms on these metrics: e.g. regret, sample complexity, computational complexity, empirical performance, convergence, etc.

CO3: Describe the exploration vs exploitation challenge and compare and contrast at least two approaches for addressing this challenge (in terms of performance, scalability, complexity of implementation, and theoretical guarantees).

Text Books:

1. Reinforcement Learning: An Introduction, Sutton and Barto, 2nd Edition, MIT Press.
2. Reinforcement Learning, Marco Wiering, Martijn Otterlo, Springer.
3. Deep Learning, Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press.

References Books:

1. Deep Reinforcement Learning,
2. Foundations of Deep Reinforcement Learning: Theory and Practice in Python (Addison-Wesley Data & Analytics Series), Laura Graesser (Author), Wah Loon Keng, Addison-Wesley Professional.
3. Reinforcement Learning: Industrial Applications of Intelligent Agents, Phil Winder, O'Reilly Media.

Course Name: **Cyber Security**
Course Code: **CS-462**
Course Type: **Stream Elective-I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the fundamental concepts and terminology of cybersecurity.
- To enable the students to identify and assess cybersecurity threats and vulnerabilities.
- To enable the students to develop skills in implementing cybersecurity measures and controls.

Unit Number	Course Content	Contact Hours
UNIT-01	**Cyber Security Concepts:** Essential Terminologies: CIA, Risks, Breaches, Threats, Attacks, Exploits. Information Gathering (Social Engineering, Foot Printing & Scanning). Port Scanners, Network scanners. Cryptography and Cryptanalysis, Symmetric key Cryptography, Asymmetric key Cryptography, Message Authentication, Digital Signatures, Applications of Cryptography. Overview of Firewalls- Types of Firewalls, User Management, VPN Security, Security Protocols, Security at Transport Layer, Security at Network Layer-IPSec.	**8L**
UNIT-02	**Server Security:** Server Security, OS Security, Physical Security, Introduction to Networks, Network packet Sniffing, Network Design Simulation. DOS/ DDOS attacks. Asset Management and Audits, Vulnerabilities and Attacks. Intrusion detection and Prevention Techniques, Host based Intrusion prevention Systems, Security Information Management, Network Session Analysis, System Integrity Validation.	**8L**
UNIT-03	**Internet Security:** Internet Security, Cloud Computing & Security, Social Network sites security, Cyber Security Vulnerabilities-Overview, vulnerabilities in software, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Authorization, Unprotected Broadband communications, Poor Cyber Security Awareness. Cyber Security Safeguards- Overview, Access control, IT Audit, Authentication. Open Web Application Security Project (OWASP), Web Site Audit and Vulnerabilities assessment	**8L**
UNIT-04	**Malware:** Types of Malware: Virus, Worms, Trojans, Rootkits, Robots, Adware's, Spywares, Ransom wares, Zombies etc., OS Hardening (Process Management, Memory Management, Task Management, Windows Registry/ services another configuration), Malware Analysis. Biometrics, Mobile Computing and Hardening on android and ios, IOT Security, Web server configuration and Security. Introduction, Basic security for HTTP Applications and Services, Basic Security for Web Services like SOAP, REST etc.	**8L**
UNIT-05	**Legal Aspects:** Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, National Cyber Security Policy 2013. Introduction to Cyber Forensics, Cyber Evidence.	**4L**
UNIT-06	**Processes and Procedures:** Documentation and Management of Crime Sense, Image Capturing and its importance, Partial Volume Image, Web Attack Investigations, Denial of Service Investigations, Internet Crime Investigations, Email Crime Investigations.	**4L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Identify vulnerabilities critical to the information assets of an organization.

CO2: Understand and apply cybersecurity ideas to define security controls to ensure confidentiality, integrity, and availability.

CO3: Critically assess cybersecurity strategies and effectively communicate information security solutions.

CO4: Diagnose and investigate cybersecurity events or crimes.

Text Books

1. William Stallings, "Cryptography and Network Security", Pearson Education/PHI, 2006.
2. V.K. Jain, "Cryptography and Network Security", Khanna Publishing House.
3. Gupta Sarika, "Information and Cyber Security", Khanna Publishing House, Delhi.

References Books

1. Atul Kahate, "Cryptography and Network Security", McGraw Hill.
2. V.K. Pachghare, "Cryptography and Information Security", PHI Learning.
3. Nina Godbole, "Information System Security", Wiley 7. Bothra Harsh, "Hacking", Khanna Publishing House, Delhi.

Course Name: **Quantum Computing**
Course Code: **CS-463**
Course Type: **Stream Elective-I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- Introduction to the fast-growing field of quantum computing.
- To Study the structural units of quantum computers of the future, forming an understanding of the differences between quantum bits and classical bits.
- To study of basic quantum logical operations and algorithms for processing quantum information.
- Mastering basic knowledge about the practical use of quantum algorithms and quantum programming skills.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Information and Computations. Characteristics of Computational Systems. Computability and Algorithms. Computational Complexity.	6L
UNIT-02	**Quantum Computation:** Classical computation versus quantum computation. Quantum Computing. The Multiverse Interpretation of Quantum Mechanics.	6L
UNIT-03	**Linear Algebra for Quantum Computing:** Review of linear algebra in the context of quantum information, Dirac's bracket notation, limitation of classical algorithms. The four postulates of quantum mechanics, qubits, quantum gates and circuits. Error correction and fault-tolerant quantum computing.	8L
UNIT-04	**Basic Quantum Algorithms:** I-Deutsch's algorithm, analysing quantum algorithms, and implementing quantum circuits via QISKIT. Basic quantum algorithms II-Simon's problem and the Bernstein -Vazirani algorithm.	5L
UNIT-05	**Advanced Quantum Algorithms:** Grover's quantum search algorithm, the BBBV Theorem, and applications of Grover's algorithm. RSA, and Shor's integer factorisation algorithm.	5L
UNIT-06	**Quantum Cryptography:** Introduction to quantum cryptography (post-quantum security, quantum key distribution). Introduction to quantum information (superdense coding, nocloning theorem, quantum teleportation) Applications (quantum money, the Elitzur-Vaidman bomb).	8L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Analyze the behaviour of basic quantum algorithms.

CO2: Implement simple quantum algorithms and information channels in the quantum circuit model.

CO3: Simulate a simple quantum error-correcting code

CO4: Understand the principles of quantum information and quantum communication:- Understand quantum teleportation and its limits.

Text Books:

1. Nielsen, M. A. (2005). Cluster-state quantum computation.
2. Nielsen, M. A., & Chuang, I. L. (2010). Quantum Computation and Quantum Information (Vol. 10th anniversary ed).
3. Hassanien, A. E., Elhoseny, M., & Kacprzyk, J. (2018). Quantum Computing:An Environment for Intelligent Large Scale Real Application.

Reference Books:

1. Sakkaris, P., & Sudhakaran, R. (2019). A Multilayer Network Approach to Quantum Computing.
2. Chris Bernhardt, Quantum Computing for Everyone (The MIT Press).
3. Riley Tipton Perry, "Quantum Computing from the Ground Up", World Scientific Publishing Ltd (2012).

Course Name: **Networked Wireless Systems**
Course Code: **CS-464**
Course Type: **Stream Elective-I**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge about the Networks, Resource sharing, Design issues for the network layers.
- To introduce the fundamental concepts related to the network architectures.
- To enable the students to understand the Data centric, hierarchical, location-based, energy efficient routing etc.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction to Networked wireless systems, Sensor network, IoT etc.** **Sensor Networks:** Introduction, applications, design issues, requirement of Sensor node architecture. Overview and motivation, Resource sharing, Design issues for the network layers Networking devices. Principles of internetworking, Tunneling, Fragmentation, Naming and addressing concepts, Hierarchical naming, Domain name system, Name resolution process, IP address classes and concept of sub netting, Classless Inter-domain routing (CIDR) and DHCP concepts,	7L
UNIT-02	**Network architecture:** The internet protocols: IP, ICMP, ARP and RARP. The design issues for the transport layer, addressing, establishing connection, flow control and multiplexing. The internet protocols: TCP and UDP, Multicast routing, Mobility in networks, Mobile IP, Emerging trends in networking. Optimization goals, evaluation metrics, network design principles. Sensor network operating systems and brief introduction to sensor network programming.	8L
UNIT-03	**Network protocols:** MAC protocols; contention based and contention free based MAC protocols, hybrid protocols and energy efficiency; need for energy efficiency and power control, Active and Passive Power Conservation Mechanisms	6L
UNIT-04	**Routing protocols:** Data centric, hierarchical, location-based, energy efficient routing etc. Sensor deployment, scheduling and coverage issues, self-configuration and topology control.	6L
UNIT-05	**Data Processing**: Querying, data collection and processing, collaborative information processing and group connectivity. Target tracking, localization and identity management. Power management, Security and privacy.	6L
UNIT-06	**Network Security**: Introduction to firewalls, intrusion detection, various network attacks, Ip and Web Security	5L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Implement and realize various privileged operations critical to the functioning of an operating system and network.
CO2: Understand the basics of network architecture.
CO3: Understand and apply various addressing and sub-netting techniques, routing protocols, etc.
CO4: Understand and apply data collection and processing concepts of sensor and IoT networks.

Text Books:

1. Wireless Sensor Networks: A Networking Perspective by Jun Zheng, Abbas Jamalipour, Wiley-IEEE Press.
2. Wireless Sensor Networks-An Information Processing Approach by Feng Zhao, Leonidas Guibas, Morgan Kauffman.
3. Wireless Sensor Networks: From Theory to Applications by Ibrahiem M. M. El Emary and S. Ramakrishnan, CRC Press.
4. Wireless sensor networks by Edgar H. Callaway, AUERBACH Publications.

Reference Books:

1. Wireless Sensor Networks: Principles and Practice by Fei Hu and Xiaojun Cao, CRC Press.
2. Protocols and Architectures for Wireless Sensor Networks by Andreas Willig and Holger Karl, Wiley Publications.
3. Cryptography and Network Security Principles and Practices, Fourth Edition, William Stallings, PHI(Pearson)

Course Name: **Blockchain Technology**
Course Code: **CS-481**
Course Type: **Stream Elective-II**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- To impart knowledge of underlying mechanism of blockchain and provides the necessary knowledge needed for starting blockchain programming.
- To introduce blockchain technologies created to realize cryptocurrency like bitcoin, ethereum.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** The consensus problem - Asynchronous Byzantine Agreement - AAP protocol and its analysis - Nakamoto Consensus on permission-less, nameless, peer-to-peer network -	5L
UNIT-02	**Models:** Abstract Models for Blockchain - GARAY model - RLA Model - Proof of Work (PoW) as random oracle - formal treatment of consistency, liveness and fairness - Proof of Stake (PoS) based Chains - Hybrid models (PoW + PoS).	6L
UNIT-03	**Cryptographic Basics for Cryptocurrency:** A short overview of Hashing, signature schemes, encryption schemes and elliptic curve cryptography.	6L
UNIT-04	**Cryptocurrency and Tokens:** Bitcoin - Wallet - Blocks - Merkley Tree - hardness of mining - transaction verifiability - anonymity - forks - double spending - mathematical analysis of properties of Bitcoin.	7L
UNIT-05	**Blockchain Platforms and Tools:** Ethereum - Ethereum Virtual Machine (EVM) - Wallets for Ethereum - Solidity - Smart Contracts - some attacks on smart contracts.	6L
UNIT-06	**Recent Trends:** Zero Knowledge proofs and protocols in Blockchain, Succinct non interactive argument for Knowledge (SNARK), pairing on Elliptic curves, Zcash.	6L

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Familiarize the functional/operational aspects of cryptocurrency ECOSYSTEM.
CO2: Understand emerging abstract models for Blockchain Technology.
CO3: Identify major research challenges and technical gaps existing between theory and practice in cryptocurrency domain Syllabus

Text Books:

1. Mastering Bitcoin, 2nd Edition by Andreas M. Antonopoulos, ISBN-13: 978-1491954386, O' Reilly Media, 2017.
2. Bitcoin and Cryptocurrency Technologies Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder.
3. Blockchain Principles and Applications in IoT, Rajdeep Chakraborty, Anupam Ghosh, Valentina Emilia Balas and Ahmed A. Elngar, Chapman & Hall.

Reference Books:

1. Blockchain Technology: Exploring Opportunities, Challenges, and Applications, Sonali Vyas, Vinod Kumar Shukla, Shaurya Gupta, Ajay Prasad, CRC Press.
2. Blockchain and Web3: Building the Cryptocurrency, Privacy, and Security Foundations of the Metaverse, Winston Ma and Ken Huang, Wiley.
3. Enterprise Strategy for Blockchain: Lessons in Disruption from Fintech, Supply Chains, and Consumer Industries (Management on the Cutting Edge), Ravi Sarathy, MIT Press.

Course Name: **Approximation Algorithms**
Course Code: **CS-482**
Course Type: **Stream Elective-II**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- Students will learn the basic definitions of approximation algorithms.
- Students will learn basic algorithmic tools used to design approximation algorithms.
- Students will learn the limits of approximation, and the basic ways of proving hardness of approximation.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** The central role of matching problems and algorithms in computer science and combinatorial optimization. Matching and edge cover problems that we will consider. Brief background material on: graphs, matroids, linear programs, parallel algorithms (shared-memory and distributed-memory models), randomized algorithms. The approximation landscape: PTAS, FPTAS, inapproximability, etc. Algorithm Engineering, high performance graph algorithms.	**6L**
UNIT-02	**Combinatorial structures:** Matroids, maximum cardinality and vertex-weighted matching. Matroid intersection, maximum matching in bipartite graphs, Matroid relaxations and k-extendible systems, edge-weighted b-matching, Submodular constraints and b-edge cover.	**5L**
UNIT-03	**Techniques for exact and approximation algorithm design:** Greedy algorithms and variants: exact algorithms for vertex-weighted matching, 1/2-approximation for edge-weighted matching, 1/3-approximation for submodular matching, 3/2-approximation for b-edge cover, Augmentation based algorithms: $(1 - 1/k)$-approximation for cardinality matching and vertex- weighted matching, Proposal making algorithms: stable matching and edge-weighted matching, Primal-dual linear programming: exact and approximation algorithms for edge-weighted matching, 3/2-, 2- and Δ-approximation algorithms for b-edge cover.	**7L**
UNIT-04	**Randomized algorithms:** Sinkhorn-Knopp approximation algorithms for cardinality matching,$2/3 -$ -approximation for edge-weighted matching, Scaling based algorithms and $1-$ epsilon-approximation for edge-weighted matching, Reduction to matching for b-edge cover problems.	**5L**
UNIT-05	**Applications of matching and edge cover:** Graph construction and sparsification for semi-supervised classification and adaptive anonymization of data, Matchings and sparse matrix computations: the Dulmage-Mendelsohn decomposition of bipartite graphs and block triangular forms of sparse matrices. The strong Hall property and predicting the nonzero structure of sparse factors. The sparse range space basis problem and the sparse null space basis problem, Network alignment via approximate edge-weighted matching, Submodular matching and edge cover: load balancing parallel computations, word alignment in natural language processing, data summarization in machine learning, applications to proteomics, etc.	**7L**
UNIT-06	**Additional Topics:** Parallel matching algorithms with logarithmic depth and linear work, Inapproximability results for graph coloring, Randomized algorithms and Luby's algorithm for maximal independent sets, Fill reduction in sparse matrix factorizations	**6L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: To apply knowledge of computing and mathematics appropriate to the discipline
CO2: To analyze a problem, and identify and define the computing requirements appropriate to its solution.
CO3: To design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
CO4: To function effectively on teams to accomplish a common goal and to use current techniques, skills, and tools necessary for computing practice.
CO5: To apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Text Books:

1. The Design of Approximation Algorithms, David P. Williamson and David B. Shmoys, Cambridge University Press.
2. Approximation Algorithms, Vijay V. Vazirani, Springer, Second Printing.
3. Complexity and Approximation: Combinatorial Optimization Problems and their Approximability Properties, G. Ausiello, P. Crescenzi, G. Gambosi et al., Springer.

Reference Books:

1. Design and Analysis of Approximation Algorithms, Ding-Zhu Du, Ker-I Ko, and Xiaodong Hu, Springer.
2. Approximation Algorithms for NP-hard problems, Dorit S, Hochbaum (ed.), PWS Publishing Company, Boston, MA,

Course Name: **Parallel Computing**
Course Code: **CS-483**
Course Type: **Stream Elective-II**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- Define terminology commonly used in parallel computing, such as efficiency and speedup.
- Describe different parallel architectures, inter-connect networks, programming models, and algorithms for common operations such as matrix-vector multiplication.
- Given a parallel algorithm, analyze its time complexity as a function of the problem size and number of processors.
- Given a parallel algorithm, an input to it, and the number of processors, show the steps performed by that algorithm on that input.
- Given a parallel algorithm, implement it using MPI, OpenMP, pthreads, or a combination of MPI and OpenMP.

Unit Number	Course Content	Contact Hours
UNIT-01	**Introduction:** Why parallel computing? Shared memory and distributed memory parallelism, Amdahl's law, speedup and efficiency, supercomputers.	**5L**
UNIT-02	**MPI Basics:** Point-to-point communication, collective communication, synchronous/ asynchronous send/recv, algorithms for gather, scatter, broadcast, reduce. OpenMP.	**5L**
UNIT-03	**Topology:** Network topologies, network evaluation metrics, communication cost, routing in interconnection networks, static and adaptive routing, process-to-processor mapping.	**6L**
UNIT-04	**Scalability Issues:** Scalability, benchmarking, performance modeling, impact of network topologies, parallel code analysis and profiling.	**7L**
UNIT-05	**Implementation:** Domain decomposition, communication-to-computation ratio, load balancing, adaptivity, case studies: weather and material simulation codes.	**7L**
UNIT-06	**Advanced Topics:** MPI I/O algorithms, contemporary large-scale I/O architecture, I/O bottlenecks. Job scheduling, RDMA, one-sided communication, NVM, extreme scale computing: issues and trends.	**6L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Be familiar with parallel architectures and be familiar with parallel algorithm design.
CO2: Be familiar with parallel performance analysis and master MPI programming.
CO3: Master multi-threaded programming and be familiar with OpenMP programming.
CO4: Be familiar with GPU/CUDA programming and be exposed to Hadoop programming and also be exposed to parallel applications.

Text Books:

1. Peter S Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2011.
2. DE Culler, A Gupta and JP Singh, Parallel Computer Architecture: A Hardware/Software Approach Morgan-Kaufmann, 1998.
3. Marc Snir, Steve W. Otto, Steven Huss-Lederman, David W. Walker and Jack Dongarra, MPI - The Complete Reference, Second Edition, Volume 1, The MPI Core.

References Books:

1. William Gropp, Ewing Lusk, Anthony Skjellum, Using MPI : portable parallel programming with the message-passing interface, 3rd Ed., Cambridge MIT Press, 2014.
2. A Grama, A Gupta, G Karypis, and V Kumar, Introduction to Parallel Computing. 2nd Ed., Addison-Wesley, 2003.

Course Name:	**Topics in Theoretical Computer Science**
Course Code:	**CS-484**
Course Type:	**Stream Elective-II**

Contact Hours/Week: **3L** Course Credits: **03**

Course Objectives

- Cover a collection of geometric techniques that apply in modern algorithm design
- Explore advanced graph theory concepts to solve complex problems
- Cover convex geometry and its role in computational optimization
- Examine iterative methods for linear algebra and analyze efficiency in solving large-scale linear algebraic problems
- Cover lattices and basis reduction techniques and their applications in cryptosystems

Unit Number	Course Content	Contact Hours
UNIT-01	**Spectral Graph Theory:** Graph Laplacians and their eigenvalues, connections to random walks and mixing, isoperimetric and Cheeger inequalities, expanders, and random graphs. Applications to include graph cutting, clustering, approximate counting, disjoint path problems, routing, and graph drawing.	**6L**
UNIT-02	**Convex Geometry:** Geometric properties of high-dimensional convex bodies, Fritz John's theorem and isotropy, Brunn-Minkowski and isoperimetric inequalities, concentration of measure and connections to probability theory. Applications to include volume computation and convex programming.	**6L**
UNIT-03	**Multiplicative Weights:** The multiplicative weights update method, its geometric meaning, and the many ways that it appears in modern computer science, with a focus on its use in optimization. Applications to include fast approximation algorithms for graph problems, "boosting" in learning and complexity theory, online algorithms, and zero-sum games.	**6L**
UNIT-04	**Iterative Methods for Linear Algebra:** How to use geometric information to quickly solve linear systems and eigenvalue problems. Will cover basic iterative methods, the Lanczos algorithm, conjugate gradients, preconditioning, and how spectral graph theory can be used to improve the construction of preconditioners.	**6L**
UNIT-05	**Lattices and Basis Reduction:** Basic properties of lattices, Minkowski's theorem, and the LLL algorithm. Applications to include solving low-dimensional integer programs and breaking cryptosystems.	**6L**
UNIT-06	**LP- and SDP-based Approximation Algorithms for NP-Hard Problems:** Linear and semidefinite programming relaxations of NP-hard problems, rounding techniques, and primal-dual methods.	**6L**

Course Outcomes

Upon successful completion of the course, the students will be able to

CO1: Understand concepts in theoretical computer science to analyze and solve real-world problems in various domains.

CO2: Implement different methods for algorithm design and analysis in theoretical computer science.

CO3: Efficiently solve large-scale linear algebraic problems arising in theoretical computer science and related fields.

Text Books:

1. Introduction to the theory of computation, Michael Sipser, Thomson Course Technology
2. Introduction to Languages and the Theory of Computation, John C. Martin, McGraw-Hill, Inc., New York.
3. The mixing rate of Markov chains, an isoperimetric inequality, and computing the volume, Lovasz, L., and M. Simonovits. Foundations of Computer Science.

Reference Books:

1. An Introduction to the Conjugate Gradient Method Without the Agonizing Pain, Shewchuk, J.
2. An Elementary Introduction to Modern Convex Geometry, Ball, K.
3. Randomized Algorithms, Motwani, Rajeev, and Prabhakar Raghavan, Cambridge, UK: Cambridge University Press.

END OF SYLLABUS