INSTITUTE OF SCIENCE, TECHNOLOGY & ADVANCED STUDIES (VISTAS)
(Deemed to be University Estd. u/s 3 of the UGC Act, 1956)
PALLAVARAM, THALAMBUR, PERIYAPALAYAM, THIRUVANMIYUR - CHENNAI

POSTGRADUATE DEGREE PROGRAMME

M.Tech
Computer Science and Engineering

CURRICULUM AND SYLLABUS

REGULATION 2026

Learning Outcomes Based Curriculum Framework (LOCF)

Effective from the Academic year

2026-2027

Department of
Computer Science and Engineering

School of Engineering

VISION OF THE DEPARTMENT
To develop a knowledge hub for Computer Science Engineers and Technocrats in application of their competence for the betterment of the Individual, Industry and Society.

MISSION OF THE DEPARTMENT	
M1	To nurture the students to be industry ready by providing a strong conceptual foundation and by enhancing their employability and entrepreneurial skills.
M2	To provide holistic growth by conducting relevant enrichment programs, which includes curricular, co-curricular, extra-curricular and extension activities.
M3	To inculcate innovation and creativity through practically viable Internships and Project works.
M4	To create a research oriented mindset and focus in fulfilling growing demands of the society through mentoring and lifelong learning.

PROGRAMME EDUCATIONAL OUTCOMES (PEO)	
PEO1	Exhibit strong Computer Science and Engineering knowledge to solve real-world problems using appropriate and emerging technologies.
PEO2	Demonstrate leadership and entrepreneurial skills in professional practice.
PEO3	Engage in continuous learning and career advancement through adaptability and professional development.
PEO4	Uphold ethical values and contribute responsibly to society.

	PROGRAMME OUTCOMES (PO)
PO1	**Engineering Knowledge**: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.
PO2	**Problem Analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)
PO3	**Design/Development of Solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)
PO4	**Conduct Investigations of Complex Problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).
PO5	**Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)
PO6	**The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).
PO7	**Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)
PO8	**Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
PO9	**Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences.
PO10	**Project Management and Finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

- PO11	**Life-Long Learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)
	PROGRAMME SPECIFIC OUTCOMES (PSO)
PSO1	An expert with an aptitude and competence to analyse, design, model,develop, test and manage information systems to offer customized solutions.
PSO2	A professional with learning abilities to face upcoming challenges in the field of Information Technology.

VISTAS

Department of Computer Science and Engineering

B.Tech.CSE Specialization in Software Engineering

Regulation 2026

CREDIT DISTRIBUTION

Minimum credits to be earned: 80

B.TECH Computer Science and Engineering					
Credits Per Semester					
Course Category	1	2	3	4	Total Credits
PCC	10	12			22
PEC	10	10	4	-	24
OEC			4	-	4
INTERNSHIP		-	2	-	2
PROJECT		-	10	16	26
MC	2	-	-	-	2
AC	-	-	-	-	0
TOTAL	22	22	20	16	80

PCC	Professional Core Courses
PEC	Professional Elective Courses
OEC	Open Elective Courses
INTERNSHIP	Internship
PROJECT	Project Work,
MC	Mandatory Courses
AC	Audit courses(non-credit)

		Hours/Week					Maximum Marks		
Category	Course Title	Lecture	Tutorial	Practical	SL	Credits	CA	SEE	Total
	SEMESTER I								
Program Core	Mathematics and Statistics for Computer Science	3	1	0	4	4	40	60	100
Program Core	Advanced Data Structures and Algorithms	3	1	0	4	4	40	60	100
Program Elective	Program Elective – I	3	1	0	4	4	40	60	100
Program Elective	Program Elective - II	3	1	0	4	4	40	60	100
Program Core	Advanced Data Structures and Algorithms Lab	0	0	4	2	2	40	60	100
Program Elective	Laboratory Course (Based on Elective – I or II)	0	0	4	2	2	40	60	100
Mandatory course	Research Methodology and IPR	2	0	0	2	2	40	60	100
Audit Course	Audit Course -I	2	0	0	0	0	0	60	-
		14	4	8	18	18			
		Hours/Week					Maximum Marks		
		Lecture	Tutorial	Practical	SL	Credits	CA	SEE	Total
	SEMESTER II								
Program Core	Advanced Algorithms	3	1	0	4	4	40	60	100
Program Core	Soft Computing	3	1	0	4	4	40	60	100
Program Elective	Program Elective - III	3	1	0	4	4	40	60	100
Program Elective	Program Elective - IV	3	1	0	4	4	40	60	100
Program Core	Advanced Algorithm and Soft Computing Lab	0	0	4	2	2	40	60	100
Program Elective	Laboratory Course (Based on Elective – III or IV)	0	0	4	2	2	40	60	100
Program Core	Mini Project	0	0	4	2	2	40	60	100
Audit Course	Audit Course -II	2	0	0	0	0	0	60	-
		12	4	12	22	22			
		Hours/Week					Maximum Marks		
		Lecture	Tutorial	Practical	SL	Credits	CA	SEE	Total
	SEMESTER III								

Program Elective V	Program Elective -V	3	1	0	4	4	40	60	100
Open Elective	Open Elective	3	1	0	4	4	40	60	100
Internship	Internship	0	0	4	2	2	40	60	100
Project	Dissertation I	0	0	20	10	10	40	60	100
		6	**2**	**24**	**20**	**20**			
		Hours/Week					Maximum Marks		
		Lecture	Tutorial	Practical		Credits	CA	SEE	Total
	SEMESTER IV								
Project	Dissertation II	0	0	32		16	40	60	100

List of Professional Electives Courses

		Hours / Week				
Code No.	Course	**L**ecture	**T**utorial	**P**ractical	**SL**	**C**redits
21MES101	Component Based Development	3	1	0	4	4
21MES102	Performance Evaluation of Systems and Networks	3	1	0	4	4
21MES103	Knowledge Engineering	3	1	0	4	4
21MES104	Visualization Techniques	3	1	0	4	4
21MES105	Infometrics	3	1	0	4	4
21MES106	Human Computer Interaction	3	1	0	4	4
21MES107	Network Engineering and Management	3	1	0	4	4
21MES108	Language Technologies	3	1	0	4	4
21MES109	Knowledge Management	3	1	0	4	4
21MES110	Integrated Software Project Management	3	1	0	4	4
21MES111	Principles of Multimedia	3	1	0	4	4
21MES112	Virtualization Techniques	3	1	0	4	4
21MES113	Service Oriented Architecture	3	1	0	4	4
21MES114	Ethical Hacking and Digital Forensics	3	1	0	4	4
21MES115	Database Tuning	3	1	0	4	4
21MES116	Enterprise Resource Planning	3	1	0	4	4
21MES117	Human Resources Management	3	1	0	4	4
21MES118	Big data Analytics	3	1	0	4	4

21MES119	Information Retrieval Techniques	3	1	0	4	4
21MES120	Social Network Analysis and Mining	3	1	0	4	4
21MES121	Data Mining Techniques	3	1	0	4	4
21MES122	Machine Learning Techniques	3	1	0	4	4
21MES123	Cloud Computing Technologies	3	1	0	4	4
21MES124	Cloud Security	3	1	0	4	4
21MES125	Cloud Storage Infrastructures	3	1	0	4	4
21MES126	Big Data Security	3	1	0	4	4
21MES127	Data Warehousing	3	1	0	4	4
21MES128	IoT Architecture and Protocols	3	1	0	4	4
21MES129	Deep Learning	3	1	0	4	4
21MES130	Blockchain And Cryptocurrency Technologies	3	1	0	4	4
21MES131	Data Center Networks	3	1	0	4	4
21MES132	Data Science With Python	3	1	0	4	4
21MES133	Intelligent Optimization Techniques	3	1	0	4	4
21MES134	Advanced Cryptography Techniques	3	1	0	4	4
21MES135	Data Analytics	3	1	0	4	4
21MES136	Applied Cryptography And Network Security	3	1	0	4	4
21MES137	Artificial Neural Networks	3	1	0	4	4
21MES138	Advanced Graph Theory	3	1	0	4	4
21MES139	Bio-Inspired Computing	3	1	0	4	4
21MES140	Sensing Techniques and Sensors	3	1	0	4	4
21MES141	Advanced Computer Architecture	3	1	0	4	4
21MES142	Object Oriented Systems Engineering	3	1	0	4	4
21MES143	Parallel Algorithms	3	1	0	4	4
21MES144	Multicore Architectures	3	1	0	4	4
21MES145	Advanced Operating Systems	3	1	0	4	4
21MES146	Pattern Recognition Techniques	3	1	0	4	4
21MES147	Advanced Database Technology	3	1	0	4	4
21MES148	Quantum Computing	3	1	0	4	4
21MES149	Edge and Fog Computing	3	1	0	4	4
21MES150	Agile Methodologies	3	1	0	4	4
21MES151	Full stack development	3	1	0	4	4

21MES152	Green Computing and Sustainability	3	1	0	4	4
21MES153	Secure Software Design And Enterprise Computing	3	1	0	4	4
21MES154	Computer Vision	3	1	0	4	4
21MES155	Distributed Data Base Systems	3	1	0	4	4
21MES156	GPU Computing	3	1	0	4	4
	Elective Laboratory Courses					
21MES157	Advanced Data Base Management System Laboratory	0	0	4	2	2
21MES158	Machine Learning Techniques Laboratory	0	0	4	2	2
21MES159	Internet Of Things Lab	0	0	4	2	2
21MES160	Computer Vision Lab	0	0	4	2	2
21MES161	Secure Software Design and Enterprise Computing Lab	0	0	4	2	2

List of Open Electives Courses

Code No.	Course	Lecture	Tutorial	Practical	SL	Credits
21MES151	Speech Processing	3	1	0	4	4
21MES152	Bio informatics	3	1	0	4	4
21MES153	ASIC Design	3	1	0	4	4
21MES154	Embedded Systems	3	1	0	4	4
21MES155	Mobile and Pervasive Computing	3	1	0	4	4
21MES156	Medical Image Processing	3	1	0	4	4
21MES157	Invehicle Intelligent Transportation System	3	1	0	4	4
21MES158	Wireless Adhoc and Sensor Networks	3	1	0	4	4
21MES159	Network Protocols	3	1	0	4	4
21MES160	Network routing Algorithms	3	1	0	4	4
21MES161	Signal Processing Techniques For Speech Recognition	3	1	0	4	4
21MES162	OFDM and MIMO Communication Systems	3	1	0	4	4
21MES163	Adaptive Signal Processing	3	1	0	4	4
21MES164	Advanced Digital Image Processing	3	1	0	4	4
21MES165	Transform Techniques and Partial Differential Equation	3	1	0	4	4
21MES166	Mobile Ad-Hoc Wireless Sensor Networks	3	1	0	4	4

21MES167	Wireless Sensor Protocols and Programming	3	1	0	4	4
21MES168	Mobile and Wireless Security	3	1	0	4	4

List of Audit Courses

		Hours / Week			
Code No.	**Course**	**Lecture**	**Tutorial**	**Practical**	Credits
21MACI11	Constitution of India	0	0	0	0
21MACI12	Safety Management	0	0	0	0
21MACI21	Disaster Management	0	0	0	0
21MACI22	Pedagogy Studies	0	0	0	0
21MACI31	Stress Management by Yoga	0	0	0	0
21MACI32	Value Education	0	0	0	0

CODE	MATHEMATICS AND STATISTICS FOR COMPUTER SCIENCE	L	T	P	SL	C
		3	1	0	4	4

COURSE OBJECTIVES

The course should enable the student to:

1. To explore graph structures and their applications.
2. To make the students, decompose the matrices into required form and to know its uses.
3. To provide information about Estimation theory, Correlation, Regression and Testing of hypothesis.
4. To enable the students to use the concepts of multivariate normal distribution and principal component

UNIT I	GRAPH STRUCTURES	12 HOURS

Graph representations – Regular Graph structures – Connectivity – Cycles – Graph coloring – Chromatic number – Chromatic partitioning – Chromatic polynomial –Vertex Covers- Independent sets – Spanning Trees – Fundamental circuits – Spanning trees in a weighted graph - Network flows.

UNIT II	ADVANCED MATRIX THEORY	12 HOURS

Eigen values using QR transformations – Generalized Eigen vectors – Canonical forms – Singular value decomposition and applications – Pseudo inverse – Least square approximations.		
UNIT III	**ESTIMATION THEORY**	**12 HOURS**
Unbiased estimators – Method of moments – Maximum likelihood estimation – Curve fitting by principles of least squares – Regression lines		
UNIT IV	**TESTING OF HYPOTHESIS**	**12 HOURS**
Sampling distributions – Type I and II errors – Test based on Normal, t, F and Chi-square distributions for testing of Mean, Variance and Proportions – Tests for independence of Attributes and Goodness of fit.		
UNIT V	**MULTIVARIATE ANALYSIS**	**12 HOURS**
Random Vectors and Matrices - Mean vectors and covariance matrices - Multivariate Normal density and its properties - Principal components - Population Principal components - Principal components for Standardized variables		

TOTAL : 60 HOURS & SELF LEARNING : 60 HOURS

TEXT BOOKS

1. Narsingh Deo., "Graph Theory with Applications to Engineering and Computer Science", Prentice-Hall of India 2014.

REFERENCES

1. Bronson, R., "Matrix Operation, Schaum's outline series", McGraw Hill, New York, Second Edition, 2011.
2. Gupta S.C. and Kapoor V.K., "Fundamentals of Mathematical Statistics", Sultan & Sons, 11th edition, 2018.
3. Richard A.Johnson and Dean W.Wichern, "Applied Multivariate Statistical Analysis", Pearson Education, Asia, 2007

CODE	ADVANCED DATA STRUCTURES AND ALGORITHMS	L	T	P	SL	C
		3	1	0	4	4

COURSE OBJECTIVES

The course should enable the student to:

1. Solve problems using different data structures and design techniques and to compare their performance, trade-offs.
2. Learn appropriate use and choice of standard data structures.
3. Develop recursive algorithms for various applications

UNIT I	**FUNDAMENTALS**	**12 HOURS**
Mathematical Induction -Asymptotic Notations (Big-O, Theta, Omega) and Properties Conditional Asymptotic Notation-Algorithm Analysis (Worst, Average, Best Case)-Introduction to Algorithmic Paradigms (Divide & Conquer, Greedy, Dynamic Programming)-Amortized Analysis-Recurrence		

Equations and Master Theorem-Introduction to Complexity Classes (P, NP – Basic Idea)-Time-Space Tradeoff		
UNIT II	**A HEAP STRUCTURES**	**12 HOURS**
Min/Max Heaps – Array-Based Representation-Heap Operations and Priority Queues-Heap-Ordered Trees and Half-Ordered Trees-d-ary Heaps and Applications-Leftist Heaps and Skew Heaps-Binomial Heaps: Structure and Operations-Introduction to Fibonacci Heaps with Applications-Applications of Heaps in Scheduling and Graph Algorithms		
UNIT III	**SEARCH STRUCTURES**	**12 HOURS**
Binary Search Trees and Applications- Self-Balancing Trees and Concept of Balancing-AVL Trees: Rotations- Red-Black Trees-B-Trees: Insert and Delete-B+ Trees and Database Indexing-Splay Trees: Basic Operations-Tries and Compressed Tries (Radix Trees)-Introduction to Hashing (Collision Resolution Techniques)		
UNIT IV	**MULTIMEDIA / SPATIAL STRUCTURES**	**12 HOURS**
Segment Trees and Applications-k-d Trees: Structure and Operations-Quad Trees: Variants and Range Queries-MX-Quad Trees-R-Trees-TV-Trees-Spatial Indexing in GIS Systems Range Searching and Nearest Neighbor Search-Introduction to Octrees		
UNIT V	**ALGORITHMS**	**12 HOURS**
Huffman Coding-Convex Hull Algorithms (Graham Scan – Basic Idea)-Topological Sort-Activity Networks and Scheduling-Flow Shop Scheduling-Counting Binary Trees-Graph Algorithms (Dijkstra / BFS / DFS – Overview)-Introduction to Approximation Algorithms Introduction to Randomized Algorithms		

TOTAL : 60 HOURS & SELF LEARNING : 60 HOURS

TEXT BOOKS

1. E. Horowitz, S.Sahni and Dinesh Mehta, Fundamentals of Data structures in C++, Uiversity Press, 2007.
2. E. Horowitz, S. Sahni and S. Rajasekaran, Computer Algorithms/C++, Second Edition, University Press, 2007.
3. V.S. Subramanian, Principles of Multimedia Database systems, Morgan Kaufman, 1998.

REFERENCES

1. G. Brassard and P. Bratley, Algorithmics: Theory and Practice, Prentice –Hall, 1988

CODE	RESEARCH METHODOLOGY AND IPR	L	T	P	SL	C
		2	0	0	2	2

COURSE OBJECTIVES

The course should enable the student to:

1. Understand research problem formulation and analyze research related information with research ethics.
2. Understanding that when IPR would take such important place in growth of individuals & nation, it is needless to emphasis the need of information about Intellectual Property Right to be promoted among students in general & engineering in particular

UNIT I	INTRODUCTION TO RESEARCH	7 HOURS

The hallmarks of scientific research – Building blocks of science in research – Concept of Applied and Basic research – Quantitative and Qualitative Research Techniques – Need for theoretical frame work – Hypothesis development – Hypothesis testing with quantitative data. Research design – Purpose of the study: Exploratory, Descriptive, Hypothesis Testing

UNIT II	EXPERIMENTAL DESIGN	7 HOURS

Laboratory and the Field Experiment – Internal and External Validity – Factors affecting Internal validity. Measurement of variables – Scales and measurements of variables. Developing scales – Rating scale and attitudinal scales – Validity testing of scales – Reliability concept in scales being developed – Stability Measures

UNIT III	DATA COLLECTION METHODS	7 HOURS

Interviewing, Questionnaires, etc. Secondary sources of data collection. Guidelines for Questionnaire Design – Electronic Questionnaire Design and Surveys. Special Data Sources: Focus Groups, Static and Dynamic panels. Review of Advantages and Disadvantages of various Data-Collection Methods and their utility. Sampling Techniques – Probabilistic and non-probabilistic samples. Issues of Precision and Confidence in determining Sample Size. Hypothesis testing, Determination of Optimal sample size

UNIT IV	DATA ANALYSIS AND RESEARCH REPORT	7 HOURS

Data Analysis – Factor Analysis – Cluster Analysis – Discriminant Analysis – Purpose of the written report – Concept of audience – Basics of written reports. Integral parts of a report – Title of a report, Table of contents, Abstract, Synopsis, Introduction, Body of a report – Experimental, Results and Discussion – Recommendations and Implementation section – Conclusions and Scope for future work.

UNIT V	INTELLECTUAL PROPERTY RIGHTS	7 HOURS

Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications. New Developments in IPR: Administration of Patent System. New developments in IPR; IPR of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, IPR and IITs.

TOTAL : 35 HOURS & SELF LEARNING : 35 HOURS

TEXT BOOKS

1. Stuart Melville and Wayne Goddard, “Research methodology: an introduction for science & engineering students’”

REFERENCES

1. Wayne Goddard and Stuart Melville, “Research Methodology: An Introduction”
2. Ranjit Kumar, 2nd Edition , “Research Methodology: A Step by Step Guide for beginners”
3. Halbert, “Resisting Intellectual Property”, Taylor & Francis Ltd ,2007.
4. Mayall , “Industrial Design”, McGraw Hill, 1992.
5. Niebel , “Product Design”, McGraw Hill, 1974.
6. Asimov , “Introduction to Design”, Prentice Hall, 1962.
7. Robert P. Merges, Peter S. Menell, Mark A. Lemley, “ Intellectual Property in New Technological Age”, 2016.

CODE	ADVANCED DATA STRUCTURES AND ALGORITHMS LABORATORY	L	T	P	SL	C
		0	0	3	2	2

COURSE OBJECTIVES

The students will learn to:

- The lab is designed to help students develop skills to design and analyse advance data structures.
- To help students identify and apply the suitable data structure for a given problem.

List of Experiments :

Implement the following using C/C++/Java

1. Write a program to perform the following operations on singly linked list.
   Creation ii) Insertion iii) Deletion iv) Traversal.
2. Write a program to perform the following operations on doubly linked list.
   Creation ii) Insertion iii) Deletion iv) Traversal in both ways
3. Write a program that implements stack (its operations) using i) Arrays ii) linked list
4. Write a programs that implements Queue (its operations) using i) Arrays ii) linked list
5. Write C program that implements the Quick sort method to sort a given list of integers in ascending order.
6. Write C program that implement the Merge sort method to sort a given list of integers in ascending order.
7. Write C program that implement the SHELL sort method to sort a given list of integers in ascending order.
8. Write a program to perform the following:
   Creating a Binary Tree of integers
   Traversing the above binary tree in preorder, inorder and postorder.
9. Write a C program to perform the following:
   Creating a AVL Tree of integers
   Traversing the above binary tree in preorder, inorder and postorder.

10. Write a C program that uses functions to perform the following:

    Creating a SplayTree of integers

    Traversing the above binary tree in preorder, inorder and postorder.

11. Write a C program to perform the following:

    Creating a B-Tree of integers

    Traversing the above binary tree in preorder, inorder and postorder.

12. Write a program to simulate various graph traversing algorithms.

CODE	ADVANCED ALGORITHMS	L	T	P	SL	C
		3	1	0	4	4

COURSE OBJECTIVES

The course should enable the student to:

1. Understand the different problems solved by using algorithmic paradigms.
2. Apply the suitable data structure for solving a problem using various strategies
3. Differentiate the complexities of a problem solved in various approaches.

UNIT I	SORTING & GRAPH	12 HOURS

Review of various sorting algorithms, topological sorting, Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkasra's), depth-first search and computation of strongly connected components, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis.

UNIT II	MATROIDS & GRAPH MATCHING	12 HOURS

Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST. Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path.

UNIT III	FLOW-NETWORKS & MATRIX COMPUTATIONS	12 HOURS

Maxflow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm. Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition.

UNIT IV	SHORTEST PATH IN GRAPHS AND MODULO REPRESENTATION OF INTEGERS/POLYNOMIALS	12 HOURS

Floyd-Warshall algorithm and introduction to dynamic programming paradigm-Optimal Binary Search Tree, 0/1 Knapsack Problem, Longest Common Subsequence, Matrix Chain Multiplication . Chinese Remainder Theorem, Conversion between base-representation and modulo-representation. Extension to polynomials. Application: Interpolation problem.

UNIT V	LINEAR PROGRAMMING APPROXIMATION ALGORITHMS	12 HOURS

Geometry of the feasibility region and Simplex algorithm NP-completeness: proof of NP-hardness and NP-completeness-Clique Problem, Vertex-Cover Problem, Subset-Sum Problem. Approximation algorithms: Introduction, Vertex-Cover Problem

TOTAL : 60 HOURS & SELF LEARNING : 60 HOURS

TEXT BOOKS

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition,The MIT Press.

REFERENCES

1. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "The Design and Analysis of Computer Algorithms", Pearson Addison-Wesley Publication.
2. Jon Kleinberg, Eva Tardos, "Algorithm Design", Pearson Addison- Wesley Publication

CODE	SOFT COMPUTING	L	T	P	SL	C
		3	1	0	4	4

COURSE OBJECTIVES

The course should enable the student to:

1. introduce soft computing concepts and techniques and foster their abilities in designing appropriate technique for a given scenario.
2. implement soft computing-based solutions for real-world problems.
3. give students knowledge of non-traditional technologies and fundamentals of artificial neural networks, fuzzy sets, fuzzy logic, genetic algorithms.

UNIT I	INTRODUCTION	12 HOURS
Evolution of Computing: Soft Computing Constituents, From Conventional AI to Computational Intelligence, Hard vs soft computing		
UNIT II	**ARTIFICIAL NEURAL NETWORKS**	**12 HOURS**
Fundamental concepts, Evolution of neural networks, Basic models of artificial neural network, Important terminologies of ANNs. McCulloch-Pitts neuron, Linear separability, Hebb network. Supervised Learning Neural Networks:Perceptron networks, Adaptive linear neuron(Adaline), Multiple Adaptive linear neuron (Madaline), Back propagation network		
UNIT III	**UNSUPERVISED LEARNING NEURAL NETWORKS**	**12 HOURS**
Unsupervised Learning Neural Networks: Kohonen self organizing networks, Adaptive resonance theory. Associate Memory Networks: Bidirectional associative memory network, Hopfield networks.		
UNIT IV	**FUZZY LOGIC**	**12 HOURS**
Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions: Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making..		
UNIT V	**GENETIC ALGORITHMS**	**12 HOURS**

Introduction to Genetic Algorithms (GA), Applications of GA in Machine Learning: Simple genetic algorithm, General genetic algorithm, Classification of genetic algorithm, Machine Learning Approach to Knowledge Acquisition.

TOTAL : 45 HOURS & SELF LEARNING : 45 HOURS

TEXT BOOKS

1. S.N. Sivanandam& S.N. Deepa, "Principles of soft computing", Wiley publications, 2nd Edition, 2008.
2. Jyh:Shing Roger Jang, Chuen:Tsai Sun, EijiMizutani, Neuro :Fuzzy and Soft Computing , Prentice:Hall of India, 2003

REFERENCES

1. LiMin Fu, "Neural Networks in Computer Intelligence" ,McGraw-Hill edition, 1994.
2. Simon Haykins, "Neural Networks a Comprehensive Foundation", PHI, second edition
3. George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications , Prentice Hall, 1995.

CODE	ADVANCED ALGORITHM and SOFT COMPUTING LAB	L	T	P	SL	C
		0	0	3	2	2

COURSE OBJECTIVES

The students will learn to:

- Familiarize with efficient utilization of programming language constructs and strategies to solve real time problems.
- Fundamentals of Neural Networks & Feed Forward Networks, Associative Memories & ART Neural Networks
- Fuzzy Logic and Fuzzy Systems; Genetic Algorithms and its design..

List of Experiments :

1. Implementation of Sorting- heap sort, quick sort, topological sort.
2. Implementation of Minimum Spanning Trees.
3. Implementation of Maximum Sub-Array Problem, Stassen's Matrix
4. Multiplication
5. Implementation of Shortest Path Algorithms.
6. Implementation of Longest Common Subsequence.
7. Implementation of Matrix Chain Multiplication, Simplex Algorithm.
8. Implementation of Simple Neural Network (McCulloh-Pitts model) for realizing AND Operation and OR operation using Perceptron learning algorithm.
9. Implementation of XOR problem using MADALINE network
10. Design and implementing the back Propagation algorithm for training a non-linear network.
11. Implementation of BAM network.

12. Implementation of KSOFM network for Clustering.
13. Implement the Genetic Algorithm for TSP.