Study Scheme of M. Tech. Computer Science and Engineering

Programme Name	M. Tech. Computer Science and Engineering
Curriculum applicable to	AY 2025-26 onwards
Curriculum Version:	1.0
Approved by / Date	BoS / 22-May-2025

SEMESTER 1st		**Contact Hours/Week**			**Maximum Marks**			**Credits**
Subject Code	**Subject Name**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
25C1CSP-101	Mathematical Foundations of Computer Science	3	0	0	40	60	100	3
25C1CSP-102	Advanced Data Structures	3	0	0	40	60	100	3
25C1CSP-111	Advanced Data Structures Lab	0	0	4	60	40	100	2
25C1CSP-PE1-XX	Programme Elective I 01-Machine Learning 02-Wireless Sensor Networks 03- Introduction to Intelligent Systems	3	0	0	40	60	100	3
25C1CSP-112	Lab Based on Electives	0	0	4	60	40	100	2
25C1CSP-PE2-XX	Programme Elective II 01-Data Science 02- Distributed systems 03-Advanced Wireless and Mobile Networks	3	0	0	40	60	100	3
25C1CCP-01	Research Methodology and IPR	2	0	0	40	60	100	2
25C1CCP-XX	Audit Course I	2	0	0	S/US*	0	0	0
Total		**16**	**0**	**8**	**320**	**380**	**700**	**18**
Total Contact Hours/Week = 24								
Total number of credits for the program will be as per guidelines given by AICTE/ UGC/ IKGPTU								

S/Us* - Satisfactory/Unsatisfactory

SEMESTER 2nd		Contact Hours/Week			Maximum Marks			Credits
Subject Code	**Subject Name**	**L**	**T**	**P**	**Int.**	**Ext.**	**Total**	
25C1CSP-201	Advanced Algorithms	3	0	0	40	60	100	3
25C1CSP-211	Lab Based on Cores	0	0	4	60	40	100	2
25C1CSP-202	Soft Computing	3	0	0	40	60	100	3
25C1CSP-PE3-XX	Programme Elective III 01 - Computer Vision 02 - Data Preparation and Analysis 03 - Secure Software Design and Enterprise Computing	3	0	0	40	60	100	3
25C1CSP-212	Lab Based on Electives	0	0	4	60	40	100	2
25C1CSP-PE4-XX	Programme Elective IV 01 - Human and Computer Interaction 02 - GPU Computing 03 - Digital Forensics	3	0	0	40	60	100	3
25C1CCP-XX	Audit Course II	2	0	0	S/US*	0	0	0
25C1CSP-203	Mini Project with Seminar	2	0	0	60	40	100	2
Total		**16**	**0**	**8**	**320**	**380**	**700**	**18**
Total Contact Hours/Week = 24								
Total number of credits for the program will be as per guidelines given by AICTE/ UGC/ IKGPTU								

S/Us* - Satisfactory/Unsatisfactory

Details of Audit courses-I

Subject Code	Subject Name
25C1CCP –02	English for Research Paper Writing
25C1CCP –03	Disaster Management
25C1CCP –04	Sanskrit for Technical Knowledge
25C1CCP –05	Stress Management by Yoga

Details of Audit courses-II

Subject Code	Subject Name
25C1CCP –06	Value Education
25C1CCP –07	Constitution of India
25C1CCP –08	Pedagogy Studies
25C1CCP –09	Personality Development through Life Enlightenment Skills

Course title	**Mathematical Foundations of Computer Science**
Course Code:	**25C1CSP-101**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	**-**
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	apply fundamental concepts of probability, analyze parametric families of distributions.
CO2	analyze sampling distributions of estimators, and applying the Method of Moments and Maximum Likelihood Estimation techniques for parameter estimation.
CO3	build the concept of permutations and combinations with and without repetition and apply the concept of graph theory to computing problems.
CO4	identify the methods of statistical inference and the role that they play in those methods.
CO5	develop the understanding of the mathematical and logical basis to modern techniques in various areas of computer science and engineering.

Detailed contents:

UNIT 1: (7 Lectures)
Probability mass, density, and cumulative distribution functions, Parametric families of distributions, Expected value, variance, conditional expectation, Applications of the univariate and multivariate Central Limit Theorem, Probabilistic inequalities, Markovchains.

UNIT 2: (7 Lectures)
Random samples, sampling distributions of estimators, Methods of Moments and Maximum Likelihood.

UNIT 3: (8 Lectures)
Statistical inference, Introduction to multivariate statistical models: regression and classification problems, principal components analysis, The problem of overfitting model assessment.

UNIT 4: (11 Lectures)
Graph Theory: Isomorphism, Planar graphs, graph coloring, Hamilton circuits and Euler cycles. Permutations and Combinations with and without repetition. Specialized techniques to solve combinatorial enumeration problems.

UNIT 5: (10 Lectures)

Computer science and engineering applications: Data mining, Network protocols, analysis of Web traffic, Computer security, Software engineering, Computer architecture, operating systems, distributed systems, Bioinformatics, Machine learning.

UNIT 6: (5 Lectures)

Recent Trends in various distribution functions in the mathematical field of computer science for varying fields like bioinformatics, soft computing, and computer vision.

Suggested Readings/Books:

1. John Vince, Foundation Mathematics for Computer Science, Springer, 2015.
2. K. Trivedi.Probability and Statistics with Reliability, Queuing, and Computer Science Applications. Wiley, 2016
3. M. Mitzenmacher and E. Upfal, Probability and Computing: Randomized Algorithms and Probabilistic Analysis, 2012
4. Alan Tucker, Applied Combinatorics, Wiley, 2014

Course title	**Advanced Data Structures**
Course Code:	**25C1CSP-102**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	**-**
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	choose appropriate data structures and algorithms and outline the concept of advanced abstract data types in algorithm design.
CO2	construct symbol tables using hashing techniques to solve problems efficiently and improve computational complexity.
CO3	classify algorithms for skip lists and trees and examine their efficiency and correctness through analytical methods.
CO4	design algorithms for text processing applications and evaluate their performance based on defined efficiency criteria.
CO5	formulate and construct algorithmic approaches to solve computational geometry problems in order to maximize performance.

Detailed contents:

UNIT 1: (7 Lectures)
Dictionaries: Definition, Dictionary Abstract Data Type, Implementation of Dictionaries. Hashing: Review of Hashing, Hash Function, Collision Resolution Techniques in Hashing, Separate Chaining, Open Addressing, Linear Probing, Quadratic, Probing, Double Hashing, Rehashing, Extendible Hashing.

UNIT 2: (5 Lectures)
Skip Lists: Need for Randomizing Data Structures and Algorithms, Search and Update Operations on Skip Lists, Probabilistic Analysis of Skip Lists, Deterministic Skip Lists.

UNIT 3: (9 Lectures)
Trees: Binary Search Trees, AVL Trees, Red Black Trees, 2-3 Trees, B-Trees, B+ Trees, Splay Trees.

UNIT 4: (12 Lectures)
Text Processing: String Operations- Basic operations & pre-processing, Brute-Force Pattern Matching, The Boyer-Moore Algorithm, The Knuth-Morris-Pratt Algorithm, Standard Tries, Compressed Tries, Suffix Tries, The Huffman Coding Algorithm, The Longest Common

Subsequence Problem (LCS), Applying Dynamic Programming to the LCS Problem.

UNIT 5: (10 Lectures)
Computational Geometry: One Dimensional Range Searching, Two-Dimensional Range Searching, constructing a Priority Search Tree, Searching a Priority Search Tree, Priority Range Trees, Quadtrees, k-D Trees.

UNIT 6: (5 Lectures)
Recent Trends in Hashing: Bloom filters, consistent hashing, Trees: Scapegoat trees, Treaps, fusion trees, and various computational geometry methods for efficiently solving the new evolving problem, Dynamic convex hull algorithm.

Suggested Readings/Books:

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 2nd Edition, Pearson, 2004.
2. M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley, 2002.
3. Peter Brass, Advanced Data Structures, Cambridge University Press, 2014

Course title	**Advanced Data Structures Lab**
Course Code:	**25C1CSP-111**
Scheme and Credits	**L T P C** **Semester – I** **0 0 4 2**
Pre-requisite (if any)	**-**
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	recall basic concepts of algorithm analysis and show how hashing techniques relate to correctness and complexity.
CO2	construct programs using various data structures such as lists and trees and make use of techniques to study their complexity.
CO3	classify algorithms used in computational geometry and examine their applications across different problem domains.
CO4	evaluate how text processing algorithms can be used to solve various real-time problems and justify their effectiveness in practical applications.
CO5	develop and propose algorithm design solutions in a project setting to creatively solve real-world problems.

Detailed contents:

Programs may be implemented using C/C++/Java

Experiment 1: Write a program to store k keys into an array of size n at the location computed using a hash function, loc = key % n, where k<=n and k takes values from [1 to m], m>n. To handle the collisions use the following collision resolution techniques,
a. Linear probing
b. Quadratic probing
c. Double hashing/rehashing
d. Chaining

Experiment 2: Write a program for Binary Search Tree to implement following operations:
a. Insertion
b. Deletion i. Delete node with only child ii. Delete node with both children
c. Finding an element
d. Finding Min element
e. Finding Max element
f. Left child of the given node
g. Right child of the given node
h. Finding the number of nodes, leaves nodes, full nodes, ancestors, descendants.

Experiment 3: Write a program for AVL Tree to implement following operations: (For nodes as integers)
a. Insertion: Test program for all cases (LL, RR, RL, LR rotation)
b. Deletion: Test Program for all cases (R0, R1, R-1, L0, L1, L-1)
c. Display: using set notation.

Experiment 4: Write a program to implement Red-Black trees with insertion and deletion operation for the given input data as Integers/Strings

Experiment 5: Write a program to implement insertion, deletion, display and search operation in m-way B tree (i.e. a non-leaf node can have at most m children) for the given data as integers.

Experiment 6: Write a program to perform string matching using Knuth-Morris-Pratt algorithm.

Experiment 7: Write a program to perform string matching using Boyer-Moore algorithm.

Experiment 8: Write a program to implement 2-D range search over computational geometry problem

Experiment 9: Write a program on latest efficient algorithms on trees for solving contemporary problems.

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Machine Learning**
Course Code:	**25C1CSP-PE1-01**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	**-**
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	recall the concept of hypothesis and its role in generalization and decision-making.
CO2	apply machine learning techniques to develop solutions for real-world problems in domains such as healthcare, finance and environmental sciences.
CO3	distinguish between supervised and unsupervised learning techniques based on their characteristics and use-cases.
CO4	evaluate the performance and suitability of neural network architectures, including perceptron's and multi-layer networks for different problem domains.
CO5	design and develop data classification and prediction models using Naïve Bayes and Bayesian Networks for real-world applications.

Detailed contents:

UNIT 1: Supervised Learning (Regression/Classification) (10 Lectures)
Basic methods: Distance-based methods, Nearest-Neighbours, Decision Trees, Naive Bayes. Linear models: Linear Regression, Logistic Regression, Generalized Linear Models. Support Vector Machines, Nonlinearity and Kernel Methods. Beyond Binary Classification: Multi-class/Structured Outputs, Ranking.

UNIT 2: Unsupervised Learning (7 Lectures)
Clustering: K-means/Kernel K-means, Hierarchical clustering. Dimensionality Reduction: PCA and kernel PCA Matrix Factorization and Matrix Completion. Generative Models (mixture models and latent factor models), Application of Association Rule Learning

UNIT 3: (6 Lectures)
Evaluating Machine Learning algorithms and Model Selection, Introduction to Statistical Learning Theory, Ensemble Methods (Boosting, Bagging, Random Forests).

UNIT 4: (9 Lectures)
Sparse Modeling and Estimation, Modeling Sequence/Time-Series Data, Deep Learning and Feature Representation Learning.

UNIT 5: (9 Lectures)
Scalable Machine Learning (Online and Distributed Learning) A selection from some other advanced topics, e.g., Semi-supervised Learning, Active Learning, Reinforcement Learning, Inference in Graphical Models, Introduction to Bayesian Learning and Inference.

UNIT 6: (5 Lectures)
Recent trends in various learning techniques of machine learning and classification methods for IOT applications. Various models for IOT applications, Genetic Algorithms.

Suggested Readings/Books:

1. Prateek Bhatia, Machine Learning with Python: Principles and Practical Techniques, 2024.
2. Rajiv Chopra, Machine Learning, Second Edition, 2020.
3. Oliver Theobald, Machine Learning for Absolute Beginners, 2024.

Course title	**Wireless Sensor Networks**
Course Code:	**25C1CSP –PE1-02**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	describe and explain radio standards and communication protocols for wireless sensor networks.
CO2	explain the function of the node architecture and use of sensors for various applications.
CO3	familiar with architectures, functions and performance of wireless sensor networks systems and platforms.

Detailed contents:

UNIT 1: (9 Lectures)
In Introduction to Wireless Sensor Networks: Course Information, Introduction to Wireless Sensor Networks: Motivations, Applications, Performance metrics, History and Design factors Network Architecture: Traditional layered stack, Cross-layer designs, Sensor Network Architecture Hardware Platforms: Motes, Hardware parameters.

UNIT 2: (9 Lectures)
Introduction to ns-3: Introduction to Network Simulator 3 (ns-3), Description of the ns-3 core module and simulation example.

UNIT 3: (8 Lectures)
Medium Access Control Protocol design: Fixed Access, Random Access, WSN protocols: synchronized, duty-cycled. Introduction to Markov Chain: Discrete time Markov Chain definition, properties, classification and analysis MAC Protocol Analysis: Asynchronous duty-cycled. X-MAC Analysis (Markov Chain)

UNIT 4: (8 Lectures)
Security: Possible attacks, countermeasures, SPINS, Static and dynamic key distribution.

UNIT 5: (10 Lectures)
Routing protocols: Introduction, MANET protocols Routing protocols for WSN: Resource-aware routing, Data-centric, Geographic Routing, Broadcast, Multicast Opportunistic Routing Analysis: Analysis of opportunistic routing (Markov Chain) Advanced topics in wireless sensor networks.

UNIT 6: (4 Lectures)
ADVANCED TOPICS : Recent development in WSN standards, software applications.

Suggested Readings/Books:

1. W. Dargie and C. Poellabauer, "Fundamentals of Wireless Sensor Networks –Theory andPractice", Wiley 2010
2. Kazem Sohraby, Daniel Minoli and Taieb Znati, "wireless sensor networks -Technology, Protocols, and Applications", Wiley Interscience 2007
3. Takahiro Hara, Vladimir I. Zadorozhny, and Erik Buchmann, "Wireless Sensor Network Technologies for the Information Explosion Era", springer 2010

Course title	**Introduction to Intelligent Systems**
Course Code:	**25C1CSP – PE1 - 03**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	demonstrate knowledge of the fundamental principles of intelligent systems
CO2	analyses and compare the relative merits of a variety of AI problem solving techniques.
CO3	apply suitable methods for searching and optimization.
CO4	integrate and implement different knowledge representation techniques.
CO5	implement learned algorithm design techniques in a project to gain hands-on experience in solving practical problems.

Detailed contents:

UNIT 1: (9 Lectures)
Biological foundations to intelligent systems I: Artificial neural networks, Back propagation networks, Radial basis function networks, and recurrent networks.

UNIT 2: (6 Lectures)
Biological foundations to intelligent systems II: Fuzzy logic, knowledge Representation and inference mechanism, genetic algorithm, and fuzzy neural networks.

UNIT 3: (7 Lectures)
Search Methods Basic concepts of graph and tree search. Three simple search methods: breadth-first search, depth-first search, iterative deepening search. Heuristic search methods: best-first search, admissible evaluation functions, hill- climbing search. Optimization and search such as stochastic annealing and genetic algorithm.

UNIT 4: (9 Lectures)
Knowledge representation and logical inference Issues in knowledge representation. Structured representation, such as frames, and scripts, semantic networks and conceptual graphs. Formal logic and logical inference. Knowledge-based systems structures, its basic components. Ideas of Blackboard architectures.

UNIT 5: (7 Lectures)
Reasoning under uncertainty and Learning Techniques on uncertainty reasoning such as Bayesian reasoning, Certainty factors and Dempster-Shafer Theory of Evidential reasoning, A study of

different learning and evolutionary algorithms, such as statistical learning and induction learning.

UNIT 6: (5 Lectures)
Recent trends in Fuzzy logic, Knowledge Representation.

Suggested Readings/Books:

1. Luger G.F. and Stubblefield W.A. (2008). Artificial Intelligence: Structures and strategies for Complex Problem Solving. Addison Wesley, 6th edition.
2. Russell S. and Norvig P. (2009). Artificial Intelligence: A Modern Approach. Prentice-Hall, 3rdedition.

Course title	**Lab Based on Electives I and II** **2 hours for Lab based on Elective I and 2 hours for Lab based on Elective II**
Program Elective Subject	**Data Science Lab**
Course Code:	**25C1CSP – 112**
Scheme and Credits	**L T P C** **Semester – I** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	apply the various data collection and data analysis techniques on different datasets.
CO2	implement K-means clustering, Linear Regression, and Logistic Regression.
CO3	implement Naïve Bayesian classifier and Decision Trees.
CO4	use Simulation tools for Data science and analysis.

Detailed contents:

The programs may be implemented using Python/MATLAB/R

Expt. 1:- Introduction to R

This Cycle introduces you to the use of the R statistical package within the Data Science and ig Data Analytics environment. After completing the tasks in this cycle you should able to:

a. Read data sets into R, save them, and examine the contents. Tasks you will complete in this Cycle include:

- Invoke the R environment and examine the R workspace.
- Created table and datasets in R.
- Examine, manipulate and save datasets.
- Exit the R environment.

Expt. 2: Basic Statistics and Visualization This Cycle introduces you to the analysis of data using the R statistical package within the Data Science and Big Data Analytics environment. After completing the tasks in Tins Cycle you should able to:

- Perform summary (descriptive) statistics on the datasets.
- Create basic visualizations using R both to support investigation of the data as well as exploration of the data.
- Create plot visualizations of the data using a graphics package.

Tasks you will complete in this Cycle include:

- Reload datasets into the R statistical package.
- Perform summary statistics on the data.

- Remove outliers from the data.
- Plot the data using R.
- Plot the data using lattice and ggplot.

Expt. 3: K-means Clustering This Cycle is designed to investigate and practice K-means Clustering. After completing the tasks in this Cycle you should able to:

- Use R functions to create K-means Clustering models.
- Use ODBC connection to the database and execute SQL statements and load datasets from the database in an R environment.
- Visualize the effectiveness of the K-means Clustering algorithm using graphic capabilities in R.
- Use the ODBC connection in the R environment to create the average house hold income from the census database as test data for K-means Clustering.
- Use R graphics functions to visualize the effectiveness of the K-means Clustering algorithm.

Expt. 4: Association Rules This Cycle is designed to investigate and practice Association Rules. After completing the tasks in this Cycle you should able to:

- Use R functions for Association Rule based models.

Tasks you will complete in this Cycle include:-

- Use the R-Studio environment to code Association Rule models.
- Apply constraints in the Market Basket Analysis methods such as minimum thresholds on support and confidence measures that can be used to select interesting rules from the set of all possible rules.
- Use R graphics "arules" to execute and inspect the models and the effect of the various thresholds.

Expt. 5: Linear Regression :- This Cycle is designed to investigate and practice linear regression. After completing the tasks in this Cycle you should able to:

- Use R functions for Linear Regression (Ordinary Least Squares - OLS).
- Predict the dependent variables based on the model.
- Investigate different statistical parameter tests that measure the effectiveness of the model.

Expt 6:- Tasks you will complete in this Cycle include:

- Use the R-Studio environment to code OLS models
- Review the methodology to validate the model and predict the dependent variable for a set of given independent variables
- Use R graphics functions to visualize the results generated with the mode

Expt. 7: Naïve Bayesian Classifier:- This Cycle is designed to investigate and practice Naive Bayesian classifier. After completing the tasks in this Cycle you should able to:

- Use R functions for Naïve Bayesian Classification
- Apply the requirements for generating appropriate training data
- Validate the effectiveness of the Naïve Bayesian Classifier with the big data

Tasks you will complete in Tins Cycle include:

- Use R-Studio environment to code the Naïve Bayesian Classifier
- Use the ODBC connection to the "census" database to create a training data set for Naïve Bayesian Classifier from the big data.
- Use the Naive Bayesian Classifier program and evaluate how well it predicts the results using the training data and then compare the results with original data.

Expt. 8: Decision Trees This Cycle is designed to investigate and practice Decision Tree (DT) models covered in the course work. After completing the tasks in This Cycle you should able to:

- Use R functions for Decision Tree models.
- Predict the outcome of an attribute base model.

Tasks you will complete in This Cycle include:

- Use the R-Studio environment to code Decision Tree Models.
- Build a Decision Tree Model based on data whose schema is composed of attributes.
- Predict the outcome of one attribute based on the model.

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Lab Based on Electives I and II** **2 hours for Lab based on Elective I and 2 hours for Lab based on Elective II**
Program Elective Subject	**Distributed Systems Lab**
Course Code:	**25C1CSP – 112**
Scheme and Credits	**L T P C** **Semester – I** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	apply the fundamental design principles and architectures of distributed systems, including concepts like client-server models, peer-to-peer networks, and cloud computing.
CO2	demonstrate the ability to perform queries on different distributed database systems.
CO3	gain practical experience by working with different distributed systems and platforms, including implementing distributed file systems, messaging systems, and cloud-based applications.
CO4	apply various fault tolerance techniques and recovery mechanisms in distributed systems, enabling them to build robust and reliable systems.

Detailed contents:
The programs may be implemented using any open source tool.

Expt. 1: Installation and configuration of database packages.

Expt. 2: Creating and managing database objects (Tables, views, indexes etc.)

Expt. 3: Creating and managing database security through user management.

Expt. 4: Creating and maintaining database links.

Expt. 5: Implement Partitioning on the database tables.

Expt.6: Implement various Transaction concurrency control methods [i.e. lock's] by executing multiple update and queries.

Expt. 7: Performance tuning of SQL queries.

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same

Course title	**Lab Based on Electives I and II** **2 hours for Lab based on Elective I and 2 hours for Lab based on Elective II**
Program Elective Subject	**Advanced Wireless and Mobile Networks Lab**
Course Code:	**25C1CSP – 112**
Scheme and Credits	**L T P C** **Semester – I** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	apply their knowledge to design and implement solutions for mobile networking scenarios, such as wireless sensor networks, ad hoc networks, and mobile IP.
CO2	analyze security, energy efficiency, mobility, scalability, and their unique characteristics in wireless networks.
CO3	compare the various network security tools as kismet and Netstumbler.
CO4	acquire the knowledge to administrate and to maintain a Wireless LAN.

Detailed contents:

The programs may be implemented using any NS2/NS3/Omnet++

Expt. 1: Setup & Configuration of Wireless Access Point (AP)

Expt.2: Study of WLAN: Ad Hoc & Infrastructure Mode

Expt. 3: Study of Bluetooth Protocol and Applications

Expt. 4: GSM modem study and SMS client-server application

Expt. 5: Mobile Internet and WML

Expt. 6: J2ME Program for Mobile Node Discovery

Expt. 7: Mobile protocol study using omnet++

Expt. 8: Wireless Network Security: kismet and Netstumbler

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Lab Based on Electives I and II** **2 hours for Lab based on Elective I and 2 hours for Lab based on Elective II**
Program Elective Subject	**Machine Learning Lab**
Course Code:	**25C1CSP – 112**
Scheme and Credits	**L T P C** **Semester – I** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	demonstrate the effective use of various machine learning tools for data analysis and model development.
CO2	design and develop Python programs to address real-time problems using machine learning algorithms.
CO3	apply suitable datasets for SVM classification and analyze results using graphical visualizations.
CO4	integrate and implement learned algorithm design techniques in a project to gain hands-on experience in solving practical problems.
CO5	design and develop innovative solutions using supervised and unsupervised learning techniques for solving real-world problems.

Detailed contents:

Programs may be implemented using WEKA/R/PYTHON etc. similar software

Experiment 1: Study of platform for Implementation of Assignments Download the open source software of your interest. Document the distinct features and functionality of the software platform. You may choose WEKA, R or any other software.

Experiment. 2: Supervised Learning – Regression: Generate a proper 2-D data set of N points. Split the data set into Training Data set and Test Data set.
i) Perform linear regression analysis with Least Squares Method.
ii) Plot the graphs for Training MSE and Test MSE and comment on Curve Fitting and Generalization Error.
iii) Verify the Effect of Data Set Size and Bias-Variance Trade off.
iv) Apply Cross Validation and plot the graphs for errors.

v) Apply Subset Selection Method and plot the graphs for errors. Describe your findings in each case.

Experiment. 3: Supervised Learning – Classification
Implement Naïve Bayes Classifier and K-Nearest Neighbour Classifier on Data set of your choice. Test and Compare for Accuracy and Precision.

Experiment. 4: Unsupervised Learning
Implement K-Means Clustering and Hierarchical clustering on proper data set of your choice. Compare their Convergence.

Experiment. 5: Dimensionality Reduction
Principal Component Analysis-Finding Principal Components, Variance and Standard Deviation calculations of principal components.

Experiment. 6: Supervised Learning and Kernel Methods
Design, Implement SVM for classification with proper data set of your choice. Comment on Design and Implementation for Linearly non-separable Dataset.

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Lab Based on Electives I and II** **2 hours for Lab based on Elective I and 2 hours for Lab based on Elective II**
Program Elective Subject	**Wireless Sensor Networks Lab**
Course Code:	**25C1CSP – 112**
Scheme and Credits	**L T P C** **Semester – I** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	design a local area network.
CO2	design, program, simulate, and experiment with sensor network software and hardware.
CO3	design a topology for network establishment.
CO4	solve various sensor network design problems
CO5	investigate design and deployment issues in wireless sensors network systems.

Detailed contents:

Programs may be implemented using NS2 / NS3

Experiment 1: Introduction to Network Simulators used for Wireless Sensor Networks

Experiment. 2: Introduction to TCL scripting: Demonstration of one small network simulator setup.

Experiment. 3: To study various trace files formats of Network Simulators.

Experiment. 4: To create a sensor network setup using the nodes configured with fixed initial energy, transmission power, reception power, routing agent, transport agent and application in rectangular area.

Experiment. 5: Create different simulation scenarios by varying MAC protocols.

Experiment. 6: Compute the performance of above created simulation scenarios of network in terms of total energy consumption, transmission latency, number of packets generated, received and dropped.

Experiment 7: To implement and compare various routing protocols using above mentioned performance metrics.
Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Lab Based on Electives I and II** **2 hours for Lab based on Elective I and 2 hours for Lab based on Elective II**
Program Elective Subject	**Introduction to Intelligent Systems Lab**
Course Code:	**25C1CSP – 112**
Scheme and Credits	**L T P C** **Semester – I** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	Implement different types of neural networks for various applications.
CO2	design and develop an intelligent system for a selected application.
CO3	develop an ability to analyze and formalize the problem (as a state space, graph, etc.) and select the appropriate search method.
CO4	apply different learning methods on various training datasets.

Detailed contents: Programs may be implemented using MATLAB/Python

Experiment 1: Implementation of simple Artificial Neural Network.

Experiment. 2: Implementation of neural network with backpropagation.

Experiment 3: Implementation of radial basis function network.

Experiment 4: Implementation of recurrent neural network.

Experiment 5: Implementation of fuzzy neural network.

Experiment 6: Implementation of iterative deepening search.

Experiment 7: Implementation of Hill climbing Search algorithm

Experiment 8: Implementation of optimization genetic algorithm.

Experiment 9: Implementation of induction-based learning method such as decision tree.

Experiment 10: Implementation of statistical learning methods such as Naive Bayes.

Course title	**Data Science**
Course Code:	**25C1CSP – PE2-01**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	interpret the knowledge and expertise to become a proficient data scientist.
CO2	make use of statistics and machine learning concepts that are vital for data science.
CO3	test for Python code to statistically analyse a dataset.
CO4	evaluate data visualizations based on their design and use for communicating stories from data.
CO5	create database connectivity with the front end.

Detailed contents:

UNIT 1: (6 Lectures)
Introduction to core concepts and technologies: Introduction, Terminology, data science process, data science toolkit, Types of data, Example applications.

UNIT 2: (7 Lectures)
Data collection and management: Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, Using multiple data sources.

UNIT 3: (10 Lectures)
Data analysis: Introduction, Terminology and concepts, Introduction to statistics, Central tendencies and distributions, Variance, Distribution properties and arithmetic, Samples/CLT, Basic machine learning algorithms, Linear regression, SVM, Naive Bayes.

UNIT 4: (11 Lectures)
Data visualization: Introduction, Types of data visualization, Data for visualization: Data types, Data encodings, Retinal variables, Mapping variables to encodings, Visual encodings.

UNIT 5: (7 Lectures)
Applications of Data Science, Technologies for visualization, Bokeh (Python).

UNIT 6: (7 Lectures)
Recent trends in various data collection and analysis techniques, various visualization techniques, application development methods of used in data science.

Suggested Readings/Books:

1. Joel Grus, Data Science from Scratch: First Principles with Python, Second Edition, O'Reilly, 2019.
2. Dr. Monika Bansal, Data Science using Python: A Step – by – Step Practical Approach for Beginners, 2022.
3. B. Uma Maheshwari, Introduction to Data Science: Practical Approach with R and Python, Wiley, 2021.

Course title	**Distributed Systems**
Course Code:	**25C1CSP – PE2 - 02**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	design trends in distributed systems and their architecture.
CO2	apply network virtualization in distributed databases.
CO3	apply remote method invocation and objects.
CO4	interpret different reliability issues in distributed databases and the failures associated with them.

Detailed contents:

UNIT 1: (8 Lectures)
Distributed data processing; What is a DDBS; Advantages and disadvantages of DDBS; Problem areas; Overview of database and computer network concepts.
Distributed Database Management System Architecture Transparencies in a distributed DBMS; Distributed DBMS architecture; Global directory issues.

UNIT 2: (11 Lectures)
Distributed Database Design:- Alternative design strategies; Distributed design issues; Fragmentation; Data allocation.
Semantics Data Control:- View management; Data security; Semantic Integrity Control.
Query Processing Issues :- Objectives of query processing; Characterization of query processors; Layers of query processing; Query decomposition; Localization of distributed data.

UNIT 3: (11 Lectures)
Distributed Query Optimization:- Factors governing query optimization; Centralized query optimization; Ordering of fragment queries; Distributed query optimization algorithms.
Transaction Management :- The transaction concept; Goals of transaction management; Characteristics of transactions; Taxonomy of transaction models.
Concurrency Control:- Concurrency control in centralized database systems; Concurrency control in DDBSs; Distributed concurrency control algorithms; Deadlock management.

UNIT 4: (8 Lectures)
RELIABILITY:- Reliability issues in DDBSs; Types of failures; Reliability techniques; Commit protocols; Recovery protocols.

UNIT 5: (6 Lectures)
Parallel Database Systems :- Parallel architectures; parallel query processing and optimization; load balancing.

UNIT 6: (4 Lectures)
Advanced Topics :- Mobile Databases, Distributed Object Management, Multi-databases.

Suggested Readings/Books:

1. Principles of Distributed Database Systems, M.T. Ozsu and P. Valduriez, Prentice-Hall, 1991.
2. Distributed Database Systems, D. Bell and J. Grimson, Addison-Wesley, 1992.

Course title	**Advanced Wireless and Mobile Networks**
Course Code:	**25C1CSP – PE2 - 03**
Scheme and Credits	**L T P C** **Semester – I** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	demonstrate advanced knowledge of networking and wireless networking and understand various types of wireless networks, standards, operations and use cases.
CO2	design WLAN, WPAN, WWAN, Cellular based upon underlying propagation and performance analysis.
CO3	demonstrate knowledge of protocols used in wireless networks and learn simulating wireless networks
CO4	design wireless networks exploring trade-offs between wire line and wireless links.
CO5	develop mobile applications to solve some of the real world problems.

Detailed contents:

UNIT 1: Introduction (11 Lectures)
Wireless Networking Trends, Key Wireless Physical Layer Concepts, Multiple Access Technologies -CDMA, FDMA, TDMA, Spread Spectrum technologies, Frequency reuse, Radio Propagation and Modelling, Challenges in Mobile Computing: Resource poorness, Bandwidth, energy etc.
Wireless Local Area Networks: IEEE 802.11 Wireless LANs Physical & MAC layer, 802.11 MAC Modes (DCF& PCF) IEEE 802.11 standards, Architecture & protocols, Infrastructure vs. Adhoc Modes, Hidden Node & Exposed Terminal Problem, Problems, Fading Effects in Indoor and outdoor WLANs, WLAN Deployment issues.

UNIT 2: Wireless Cellular Networks (10 Lectures)
1G and 2G, 2.5G, 3G, and 4G, Mobile IPv4, Mobile IPv6, TCP over Wireless Networks, Cellular architecture, Frequency reuse, Channel assignment strategies, Handoff strategies, Interference and system capacity, Improving coverage and capacity in cellular systems, Spread spectrum Technologies.

UNIT 3: (8 Lectures)
WiMAX (Physical layer, Media access control, Mobility and Networking), IEEE 802.22 Wireless Regional Area Networks, IEEE 802.21 Media Independent Handover Overview WIRELESS

SENSOR NETWORKS. Introduction, Application, Physical, MAC layer and Network Layer, Power Management, Tiny OS Overview.

UNIT 4: Wireless Pans (4 Lectures)
Bluetooth AND Zigbee, Introduction to Wireless Sensors.

UNIT 5: Security (10 Lectures)
Security in wireless Networks Vulnerabilities, Security techniques, Wi-Fi Security, DoS in wireless communication.

UNIT 6: Advanced Topics (5 Lectures)
IEEE 802.11x and IEEE 802.11i standards, Introduction to Vehicular Adhoc Networks.

Suggested Readings/Books:

1. Schiller J., Mobile Communications, Addison Wesley 2000
2. Stallings W., Wireless Communications and Networks, Pearson Education 2005
3. Stojmenic Ivan, Handbook of Wireless Networks and Mobile Computing, John Wiley and Sons Inc 2002
4. Yi Bing Lin and Imrich Chlamtac, Wireless and Mobile Network Architectures, John Wiley and Sons Inc 2000
5. Pandya Raj, Mobile and Personal Communications Systems and Services, PHI 2000

Course title	**Research Methodology and IPR**
Course Code:	**25C1CCP-01**
Scheme and Credits	**L T P C** **Semester – I** **0 0 2 2**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	outline the key steps involved in formulating a research problem based on literature and defined objectives.
CO2	identify the significance of research-related information and make use of it to define a research problem.
CO3	examine the structure of a well-organized research paper and analyze the effectiveness of scientific presentations following academic standards.
CO4	evaluate the different components of Intellectual Property Rights (IPR) and compare the processes involved in filing them.
CO5	formulate and elaborate on the scope and implications of patents and intellectual rights in protecting research outcomes.

Detailed contents:

UNIT 1:
Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations

UNIT 2:
Effective literature studies approaches, analysis Plagiarism, Research ethics.

UNIT 3:
Effective technical writing, how to write report, Paper Developing a Research Proposal, Format of research proposal, a presentation and assessment by a review committee.

UNIT 4:
Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of Patenting and Development: technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property. Procedure for grants of patents, Patenting under PCT.

UNIT 5:
Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications.

UNIT 6:
New Developments in IPR: Administration of Patent System. New developments in IPR; IPR. of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, PR and IITs.

Suggested Readings/Books:

1. Stuart Melville and Wayne Goddard, "Research methodology: an introduction for science & engineering students", 2004
2. Wayne Goddard and Stuart Melville, "Research Methodology: An Introduction", 2004
3. Ranjit Kumar, 2nd Edition , "Research Methodology: A Step by Step Guide for beginners", 2023
4. Halbert, "Resisting Intellectual Property", Taylor & Francis Ltd, 2007.
5. Mayall , "Industrial Design", McGraw Hill, 1992.
6. Niebel , "Product Design", McGraw Hill, 1974.
7. Asimov, "Introduction to Design", Prentice Hall, 1962.
8. Robert P. Merges, Peter S. Menell, Mark A. Lemley, " Intellectual Property in New Technological Age", 2016.
9. T. Ramappa, "Intellectual Property Rights Under WTO", S. Chand, 2008

Course title	**Audit Course I: English for Research Paper Writing**
Course Code:	**25C1CCP-02**
Scheme and Credits	**L T P C** **Semester – I** **2 0 0 0**
Pre-requisite (if any)	
Internal Marks	**0**
External Marks	**0**
Total Marks	**0**

Course Outcomes:

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

CO1	explain the principles of effective academic writing and how to improve writing skills.
CO2	identify and apply techniques to enhance the readability and clarity of written content.
CO3	develop the ability to write each section of a research paper with appropriate structure and coherence.
CO4	analyze the essential elements and techniques required for writing a high-quality academic paper.
CO5	apply creative thinking and apply critical thinking strategies in academic and technical writing.

Detailed contents:

UNIT 1: (4 Hours)
Planning and Preparation, Word Order, Breaking up long sentences, Structuring. Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness.

UNIT 2: (4 Hours)
Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticising, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts. Introduction.

UNIT 3: (4 Hours)
Review of the Literature, Methods, Results, Discussion, Conclusions, The Final Check.

UNIT 4: (4 Hours)
Key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature.

UNIT 5: (4 Hours)

Skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions.

UNIT 6: (4 Hours)
Useful phrases, how to ensure paper is as good as it could possibly be the first- time submission.

Suggested Readings/Books:

1. Goldbort R (2006) Writing for Science, Yale University Press (available on Google Books). 2. Day R (2006) How to Write and Publish a Scientific Paper, Cambridge University Press.
3. Highman N (1998), Handbook of Writing for the Mathematical Sciences, SIAM. Highman's book.
4. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011.

Course title	**Advanced Algorithms**
Course Code:	**25C1CSP-201**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	outline advanced algorithms and algorithmic techniques to illustrate their behaviour in solving complex computational problems.
CO2	select and apply algorithmic strategies to solve problems in different domains and measure their correctness and efficiency.
CO3	examine the principles of computational intractability, classify approaches to handle hard problems and inspect the effectiveness of approximate and heuristic techniques.
CO4	evaluate the mathematical principles supporting algorithm design and justify their implementation to enhance computational efficiency.
CO5	develop innovative solutions by combining recent advances in data structures and algorithms to address real-world problems using modern computational paradigms.

Detailed contents:

UNIT 1: (6 Lectures)
Sorting: Review of various sorting algorithms, topological sorting. Graph: 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 2: (8 Lectures)
Matroids: Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST, Kruskal's and Prim's. Graph Matching: Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path.

UNIT 3: (9 Lectures)
Flow-Networks: Maxflow -mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.
Matrix Computations: 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 4: (10 Lectures)
Shortest Path in Graphs: Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming.
Modulo Representation of integers/polynomials: Chinese Remainder Theorem, Conversion between base-representation and modulo-representation. Extension to polynomials. Application: Interpolation problem.
Discrete Fourier Transform (DFT): In complex field, DFT in modulo ring. Fast Fourier Transform algorithm. Schonhage-Strassen Integer Multiplication algorithm.

UNIT 5: (10 Lectures)
Linear Programming: Geometry of the feasibility region and Simplex algorithm.
NP-completeness: Examples, proof of NP-hardness and NP-completeness.
One or more of the following topics based on time and interest. Approximation algorithms, Randomized Algorithms, Interior Point Method, Advanced. Number Theoretic Algorithm.

UNIT 6: (5 Lectures)
Recent Trends in problem solving paradigms using recent searching and sorting techniques (Bitonic sort, sample sort) by applying recently proposed data structures. Graph-based: Compressed sparse row (CSR), Compressed Sparse Column (CSC), Graph Neural Networks (GNNs).

Suggested Readings/Books:

1. "Introduction to Algorithms" by Cormen, Leiserson, Rivest, Stein, 2009
2. "The Design and Analysis of Computer Algorithms" by Aho, Hopcroft, Ullman, 2017
3. "Algorithm Design" by Kleinberg and Tardos, 2005.

Course title	**Lab Based on Cores – Advanced Algorithms and Soft Computing**
Course Code:	**25C1CSP-211**
Scheme and Credits	**L T P C** **Semester – II** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	describe classical and advanced graph algorithms such as Dijkstra's, Floyd-Warshall, Bellman-Ford, BFS, DFS, and Edmonds' Blossom algorithm, and illustrate their applications in solving real-world problems.
CO2	identify fundamental and advanced data structures such as stacks, queues, lists, and binary search trees, and make use of both array and linked list representations.
CO3	examine divide-and-conquer and randomized strategies to analyze the efficiency of algorithms for sorting and matrix operations.
CO4	evaluate optimization problems using techniques from linear programming and approximation algorithms, and assess the effectiveness of the solutions.
CO5	formulate and design expression transformations and numerical computations using modular arithmetic, stack-based expression parsing, and Lower Upper decomposition.

Detailed contents:

Programs may be implemented using C/C++/JAVA

Experiment 1: WAP to implement Dijkstra's algorithm for single-source shortest path in a weighted directed graph.

Experiment 2: WAP to find all-pairs shortest path using Floyd-Warshall algorithm.

Experiment 3: WAP to find inverse of a triangular matrix using divide and conquer strategy.

Experiment 4: WAP to convert base (decimal/hexa) representation to modulo representation.

Experiment 5: WAP to implement FFT (Fast Fourier transform).

Experiment 6: WAP to use recursive and non-recursive function and implement Breadth First Search and Depth First Search in a graph.

Experiment 7: WAP to implement Bellman ford for single source shortest path in a weighted directed graph.

Experiment 8: WAP to implement Neural Network Tool Box by using Library functions.

Experiment 9: WAP to implement Fuzzy Logic Tool Box by using Library functions.

Experiment 10: WAP to perform operations on fuzzy sets.

Experiment 11: WAP to compute a maximum weight maximal independent set using Matroids.

Experiment 12: WAP to visualize the feasibility region of a Linear Programming Problem (LPP) with two variables.

Course title	**Soft Computing**
Course Code:	**25C1CSP – 202**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	outline soft computing methodologies for engineering problem.
CO2	apply and design fuzzy logic and reasoning to handle uncertainty and solve engineering problems.
CO3	analyze the fundamental theory and concepts of neural networks, Identify different neural network architectures, algorithms, applications and their limitations to evaluate solutions by various soft computing approaches.
CO4	examine and formulate genetic algorithm to combinatorial optimization problems.
CO5	implementation of the different methods using tools and libraries.

Detailed contents:

UNIT 1: (7 Lectures)
Introduction To Soft Computing And Neural Networks: Evolution of Computing: Soft Computing Constituents, From Conventional AI to Computational Intelligence: Machine Learning Basics.

UNIT 2: (8 Lectures)
Fuzzy Logic: 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 3: (10 Lectures)
Neural Networks: Machine Learning Using Neural Network, Adaptive Networks, Feed forward Networks, Supervised Learning Neural Networks, Radial Basis Function Networks : Reinforcement Learning, Unsupervised Learning Neural Networks, Adaptive Resonance architectures, Advances in Neural networks.

UNIT 4: (5 Lectures)
Genetic Algorithms: Introduction to Genetic Algorithms (GA), Applications of GA in Machine Learning: Machine Learning Approach to Knowledge Acquisition.

UNIT 5: (13 Lectures)
Matlab/Python Lib: Introduction to Matlab/Python, Arrays and array operations, Functions and Files, Study of neural network toolbox and fuzzy logic toolbox, Simple implementation of Artificial Neural Network and Fuzzy Logic.

UNIT 6: (5 Lectures)
Recent Trends in deep learning, various classifiers, neu ral networks and genetic algorithm. Implementation of recently proposed soft computing techniques.

Suggested Readings/Books:

1. Jyh Shing Roger Jang, Chuen Tsai Sun, Eiji Mizutani, Neuro Fuzzy and Soft Computing17, Prentice Hall of India, 2003.
2. George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications17, Prentice Hall, 1995.
3. MATLAB Toolkit Manual

Course title	**Computer Vision**
Course Code:	**25C1CSP – PE3-01**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	summarize the concepts of image formation, measurement, and analysis.
CO2	analyse various image detection and segmentation techniques.
CO3	utilize dimensionality reduction methods for feature extraction.
CO4	apply several clustering and classification algorithms using different classifier techniques.
CO5	examine the recent trends in image processing and its applications.

Detailed contents:

UNIT 1: (8 Lectures)
Overview, computer imaging systems, lenses, Image formation and sensing, Image analysis, pre-processing and Binary image analysis.

UNIT 2: (9 Lectures)
Edge detection, Edge detection performance, Hough transform, corner detection.

UNIT 3: (9 Lectures)
Segmentation, Morphological filtering, Fourier transform.

UNIT 4: (9 Lectures)
Feature extraction, shape, histogram, color, spectral, texture, using CVIP tools, Feature analysis, feature vectors, Feature Matching, distance /similarity measures, data pre- processing.

UNIT 5: (9 Lectures)
Pattern Analysis: 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, and Nonparametric methods

UNIT 6: (4 Lectures)
Deep Neural Networks, Recent trends inActivity Recognition, computational photography, Biometrics.

Suggested Readings/Books:

1. Computer Vision: Algorithms and Applications by Richard Szeliski, Springer, 2022.
2. Deep Learning, by Good fellow, Bengio, and Courville, MIT Press, 2016
3. Dictionary of Computer Vision and Image Processing by Fisher et al, Wiley 2016

Course title	**Data Preparation and Analysis**
Course Code:	**25C1CSP –PE3-02**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	apply different techniques of data gathering and preparation of data for analysis.
CO2	analyse the data for cleaning and consistency checking.
CO3	extract the data for performing the analysis.
CO4	perform different visualization techniques on the input data.

Detailed contents:

UNIT 1: (9 Lectures)
Data Gathering and Preparation: Data formats, parsing and transformation, Scalability and real-time issues.

UNIT 2: (11 Lectures)
Data Cleaning: Consistency checking, Heterogeneous and missing data, Data Transformation and segmentation.

UNIT 3: (13 Lectures)
Exploratory Analysis: Descriptive and comparative statistics, Clustering and association, Hypothesis generation.

UNIT 4: (15 Lectures)
Visualization: Designing visualizations, Time series, Geolocated data, Correlations and connections, Hierarchies and networks, interactivity.

Suggested Readings/Books:

1. Making sense of Data: A practical Guide to Exploratory Data Analysis and Data Mining, by Glenn J. Myatt, 2014

Course title	**Secure Software Design and Enterprise Computing**
Course Code:	**25C1CSP –PE3-03**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	differentiate between various software vulnerabilities.
CO2	use of Software process vulnerabilities for an organization
CO3	monitor resources consumption in a software.
CO4	interrelate security and software development process.

Detailed contents:

UNIT 1: (8 Lectures)

Secure Software Design: Identify software vulnerabilities and perform software security analysis, Master security programming practices, Master fundamental software security design concepts, Perform security testing and quality assurance.

UNIT 2: (11 Lectures)
Enterprise Application Development Describe the nature and scope of enterprise software applications, Design distributed N- tier software application, Research technologies available for the presentation, business and data tiers of an enterprise software application, Design and build a database using an enterprise database system, Develop components at the different tiers in an enterprise system, Design and develop a multi-tier solution to a problem using technologies used in enterprise system, Present software solution.

UNIT 3: (8 Lectures)
Enterprise Systems Administration Design, implement and maintain a directory-based server infrastructure in a heterogeneous systems environment, Monitor server resource utilization for system reliability and availability, Install and administer network services (DNS/DHCP/Terminal Services/Clustering/Web/Email).

UNIT 4: (8 Lectures)
Obtain the ability to manage and troubleshoot a network running multiple services, Understand the requirements of an enterprise network and how to go about managing them.

UNIT 5: (9 Lectures)
Handle insecure exceptions and command/SQL injection, Defend web and mobile applications against attackers, software containing minimum vulnerabilities and flaws.

UNIT 6: (4 Lectures)
Case study of DNS server, DHCP configuration and SQL injection attack.

Suggested Readings/Books:

1. Theodor Richardson, Charles N Thies, Secure Software Design, Jones & Bartlett, 2013
2. Kenneth R. van Wyk, Mark G. Graff, Dan S. Peters, Diana L. Burley, Enterprise Software Security, Addison Wesley, 2015

Course title	**Lab Based on Electives III and IV** **2 hours for Lab based on Elective III and 2 hours for Lab based on Elective IV**
Program Elective Subject	**Computer Vision Lab**
Course Code:	**25C1CSP – 212**
Scheme and Credits	**L T P C** **Semester – II** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	apply fundamental computer vision techniques such as image filtering, edge detection, and transformations.
CO2	contrast between image detection and segmentation algorithms (e.g., thresholding, region-based methods).
CO3	develop object and scene recognition methods using feature descriptors and classification techniques.
CO4	evaluate existing deep neural network models (e.g., CNNs) for image classification and object detection tasks.
CO5	interpret the performance of computer vision models using evaluation metrics like precision, recall, and IoU.

Detailed contents:

Programs may be implemented using MATLAB/C/C++/Java/Python on binary/grayscale/color images.

Experiment 1: Implementation of basic image transformations:
a. Log
b. Power law
c. Negation

Experiment 2: Implementation the following:
a. Histogram processing
b. Histogram equalization/matching

Experiment 3: Implementation of piecewise linear transformations a. Contrast stretching b. Grey level slicing c. Bit plane slicing

Experiment 4: Implementation of image enhancement/smoothing using
a. Linear (weighted and non-weighted filters)
b. Order statistics filters (Nonlinear filters) i. Mean ii. Median iii. Min iv. Max v. Average

Experiment 5: Implementation of image enhancement/sharpening using
a. Laplacian operators
b. Sobel's operators c. Robert's cross operators

Experiment 6: Implement the 2D-DFT to obtain Fourier coefficients and reconstruct the image, i.e., IDFT.

Experiment 7: Implement image enhancement using Fourier low pass filters
a. Ideal
b. Butterworth
c. Gaussian

Experiment 8: Implement image enhancement using Fourier high pass filters
a. Ideal
b. Butterworth
c. Gaussian

Experiment 9: Implement algorithms to detect the following in an image
a. Point
b. Line
c. Boundary

Experiment 10: Implement Hough transform to detect a line.

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Lab Based on Electives III and IV** **2 hours for Lab based on Elective III and 2 hours for Lab based on Elective IV**
Program Elective Subject	**Data Preparation and Analysis Lab**
Course Code:	**25C1CSP – 212**
Scheme and Credits	**L T P C** **Semester – II** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	differentiate between numerical and categorical attributes and apply various pre-processing techniques to clean any chosen dataset.
CO2	apply exploratory data analysis techniques to develop meaningful data visualizations.
CO3	apply Association Rule mining technique to explore relationships among various attributes.
CO4	create applications to deal with interactive datasets suitable to explore the significance of variables.

Detailed contents:

Programs may be implemented using Weka

Experiment 1: Using weka tool to explore the data.

Experiment 2: Using weka tool to do Parametric–Means.

Experiment 3: Using weka tool to do Parametric-T-Test.

Experiment 4: Using weka tool to do Correlation analysis.

Experiment 5: Preprocess the given data using weka tool.

Experiment 6: Apply different classification techniques to classify the given data set.

Experiment 7: Apply various clustering techniques to cluster the data.

Experiment 8: Apply various association rule mining algorithms.

Experiment 9: Implement classification using Decision tree.

Experiment 10: Apply Visualization methods using weka tool

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same

Course title	**Lab Based on Electives III and IV** **2 hours for Lab based on Elective III and 2 hours for Lab based on Elective IV**
Program Elective Subject	**Secure Software Design and Enterprise Computing Lab**
Course Code:	**25C1CSP – 212**
Scheme and Credits	**L T P C** **Semester – II** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	develop a security model for any enterprise-based application on its threats and vulnerabilities.
CO2	implement methodologies and tools to design secure software enterprise application.
CO3	implement the various security algorithms to be implemented for secured computing and computer networks.
CO4	compare different types of threats and attacks.
CO5	evaluate various methods of authentication and access control for web-based applications.

Detailed contents:

Experiment 1: Write a program to implement authentication to prevent various attacks.

Experiment 2: Write a program to Limit or increasingly delay failed login attempts.

Experiment 3: Create a scenario to test authentication of various security attacks.

Experiment 4: Write a program to debug backdrop entry of given source code.

Experiment 5: Write a program to debug HTTP headers, input fields, hidden fields, drop down lists, and other web components.

Experiment 6: Write a program to test Input filtering via white list validation.

Experiment 7: Create a scenario to Set Up Your Own Private Cloud Storage.

Experiment 8: Setup and configuration Various network services (DNS/ DHCP/ Terminal Services/ Clustering/ Web/ Email).

Experiment 9: Design and build a database using an enterprise database system.

Experiment 10: Design and implement a directory-based server infrastructure in a heterogeneous systems environment.

Experiment 11: An attacker wishing to execute SQL injection manipulates a standard SQL query to exploit non-validated input vulnerabilities in a database. Show different ways that this attack vector can be executed.

Experiment 12: Install IBM Rhapsody Tool using NetBeans for Java and Junit (a unit testing tool).

Experiment 13: Create a Unified Modelling Language (UML) Class diagram and a UML Sequence diagram using IBM's Rhapsody modelling tool.

Experiment 14: Configure NetBeans to use JUnit and test code written for the classes and methods described in the UML.

Course title	**Lab Based on Electives III and IV** **2 hours for Lab based on Elective III and 2 hours for Lab based on Elective IV**
Program Elective Subject	**Digital Forensics Lab**
Course Code:	**25C1CSP – 212**
Scheme and Credits	**L T P C** **Semester – II** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	apply fundamental digital forensics techniques and attributes of file systems and storage media.
CO2	demonstrate the ability to perform basic forensic data acquisition and analysis using computer and network- based applications and utilities.
CO3	evaluate the ability to accurately document forensic procedures and results.
CO4	interpret the need of workstation for the applying digital forensic techniques.

Detailed contents: Any software from the following can be used for
1.SysInternals Suite Microsoft System utilities for diagnosis of Windows systems, SANSSIFT
2. SANS Investigate Forensic Toolkit (SIFT)
3. Wireshark Network protocol analyzer
4. Trinity Rescue Kit ALinux based recovery and repair toolkit for Windows computers.
5. Kali Linux A PenTest toolkit based on Linux.

This should only be used to check your own equipment or equipment you have been asked to test

Expt. 1: To Develop multifaceted cyber-crime scenario (cyber-crime and cyber-terrorism)

- Build a top-down systematic process
- Structure the team and players
- Use an integrated Framework (SI-FI)
- Integrate GOTS, COTS, and R&D Tools
- Use real investigators / compliment with technology experts
- Carefully collect all data, decisions actions during experiment
- Develop metrics for evaluation that match scenario
- Quantify results

Expt. 2: To perform packet-level analysis using appropriate tools (e.g., Wireshark, tcpdump).

Expt. 3: To identify and extract data of forensic interest in diverse media (i.e., media forensics).

Expt. 4: To identify, modify, and manipulate applicable system components within Windows, UNIX, or Linux (e.g., passwords, user accounts, files).

Expt. 5: To collect, process, package, transport, and store electronic evidence to avoid alteration, loss, physical damage, or destruction of data.

Expt. 6: To set up a forensic workstation.

Expt. 7: To use forensic tool suites (e.g., EnCase, Sleuthkit, FTK).

Expt. 8: To use virtual machines. (e.g., Microsoft Hyper-V, VMWare vSphere, Citrix XenDesktop/Server, Amazon Elastic Compute Cloud, etc.).

Expt. 9: To conduct forensic analysis in multiple operating system environments (e.g., mobile device systems).

Expt. 10: To analyze captured malicious code (e.g., malware forensics).

Expt. 11: To use binary analysis tools (e.g., Hexedit, command code xxd, hexdump).

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Lab Based on Electives III and IV** **2 hours for Lab based on Elective III and 2 hours for Lab based on Elective IV**
Program Elective Subject	**Human and Computer Interaction Lab**
Course Code:	**25C1CSP – 212**
Scheme and Credits	**L T P C** **Semester – II** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	apply HCI and principles to interaction design.
CO2	design certain tools for blind or PH people.
CO3	apply different menus, icons for different kinds of windows.
CO4	create applications using interactive graphical interfacing with screen complexity.

Detailed contents:

Programs may be implemented using C++ / Python

Experiment 1: To understand the trouble of interacting with computers – redesign interfaces of home appliances.

Experiment 2: Design a system based on user – centered approach.

Experiment 3: Understand the principles on good screen design.

Experiment 4: Redesign existing Graphical User Interface with screen complexity.

Experiment 5: Implementation of different kinds of Menus.

Experiment 6: Implementation of different kinds of Windows.

Experiment 7: Design a system with proper guidelines for icons.

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

Course title	**Lab Based on Electives III and IV** **2 hours for Lab based on Elective III and 2 hours for Lab based on Elective IV**
Program Elective Subject	**GPU Computing Laboratory**
Course Code:	**25C1CSP – 212**
Scheme and Credits	**L T P C** **Semester – II** **0 0 4 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	understand the concepts of General Purpose GPU Programming.
CO2	interpret the architecture of GPU using different tools.
CO3	implement the CUDA environment using various matrix tools.
CO4	compare CUDA environment on single and multiple GPUs.

Detailed contents: Programs may be implemented using C.

Experiment 1: Setting up Cuda environment.

Experiment 2: Program for parallel matrix multiplication with Cuda.

Experiment 3: Program to demonstrate grids, blocks and threads.

Experiment 4: Program for parallel radix sort.

Experiment 5: Demonstrate parallel reduction with Cuda.

Experiment 6: Program to demonstrate parallel programming for merging two lists.

Experiment 7: Program to demonstrate concept of global memory.

Experiment 8: Program to demonstrate concept of multi – GPUs.

Experiment 9: Program to demonstrate concept of profiling with parallel Nsight.

Experiment 10: Implementation of deep networks for image classification with GPU Programming.

Course title	**Human and Computer Interaction**
Course Code:	**25C1CSP – PE4-01**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	understand the structure of models and theories of human computer interaction and vision.
CO2	identify and apply the guidelines for user interface.
CO3	examine the architecture of mobile Human Computer interaction.
CO4	design an interactive web interface on the basis of models studied.
CO5	apply emerging trends in HCI such as multimodal systems, speech recognition, and translation technologies.

Detailed contents:

UNIT 1: (9 Lectures)
Human: I/O channels – Memory – Reasoning and problem solving; The computer: Devices – Memory – processing and networks; Interaction: Models– frameworks – Ergonomics – styles – elements – interactivity- Paradigms.

UNIT 2: (12 Lectures)
Interactive Design basics – process – scenarios – navigation – screen design – Iteration and prototyping. HCI in software process – software life cycle – usability engineering –Prototyping in practice – design rationale. Design rules– principles, standards, guidelines, rules. Evaluation Techniques – Universal Design.

UNIT 3: (8 Lectures)
Cognitive models –Socio-Organizational issues and stake holder requirements Communication and collaboration models-Hypertext, Multimedia and WWW.

UNIT 4: (8 Lectures)
Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools.

UNIT 5: (8 Lectures)
Designing Web Interfaces – Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow. Case Studies

UNIT 6: (3 Lectures)
Recent Trends: Speech Recognition and Translation, Multimodal System.

Suggested Readings/Books:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, "Human Computer Interaction", 3rdEdition, Pearson Education, 2004 (UNIT I , II & III)
2. Brian Fling, "Mobile Design and Development", First Edition , O'Reilly Media Inc., 2009 (UNIT–IV)
3. Bill Scott and Theresa Neil, "Designing Web Interfaces", First Edition, O'Reilly, 2009.(UNIT-V).

Course title	**GPU Computing**
Course Code:	**25C1CSP – PE4-02**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	understand the concepts in parallel programming.
CO2	implement the programs on Graphic Processing Units.
CO3	synchronize CPU and GPU systems
CO4	debug and profile parallel programs.

Detailed contents:

UNIT 1: (13 Lectures)
Introduction: History, Graphics Processors, Graphics Processing Units, GPGPUs. Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel programming, CUDA OpenCL / OpenACC, Hello World Computation Kernels, Launch parameters, Thread hierarchy, Warps/ Wavefronts, Thread blocks / Workgroups, Streaming multiprocessors, 1D / 2D/ 3D thread mapping, Device properties, Simple Programs.

UNIT 2: (7 Lectures)
Memory: Memory hierarchy, DRAM / global, local / shared, private / local, textures, Constant Memory, Pointers, Parameter Passing, Arrays and dynamic Memory, Multi- dimensional Arrays, Memory Allocation, Memory copying across devices, Programs with matrices, Performance evaluation with different memories.

UNIT 3: (10 Lectures)
Synchronization: Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU Functions: Device functions, Host functions, Kernels functions, Using libraries (such as Thrust), and developing libraries.

UNIT 4: (8 Lectures)
Support: Debugging GPU Programs. Profiling, Profile tools, Performance aspects Streams: Asynchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based- Synchronization - Overlapping data transfer and kernel execution, pitfalls.

UNIT 5: (5 Lectures)
Case Studies: Image Processing, Graph algorithms, Simulations, Deep Learning.

UNIT 6: (5 Lectures)
Advanced topics: Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing.

Suggested Readings/Books:

1. Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-mei Hwu; Morgan Kaufman; 2010 (ISBN: 978-0123814722)
2. CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman; 2012 (ISBN: 978-0124159334)

Course title	**Digital Forensics**
Course Code:	**25C1CSP – PE4-03**
Scheme and Credits	**L T P C** **Semester – II** **3 0 0 3**
Pre-requisite (if any)	
Internal Marks	**40**
External Marks	**60**
Total Marks	**100**

Course Outcomes:

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

CO1	understand relevant legislation and codes of ethics.
CO2	apply computer forensics and digital detective and various processes, policies and procedures.
CO3	interpret e-discovery, guidelines and standards, evidence, tools and environment.
CO4	design web forensics and network forensics.

Detailed contents:

UNIT 1: (9 Lectures)
Digital Forensics Science: Forensics science, computer forensics, and digital forensics. Computer Crime: Criminalistics as it relates to the investigative process, analysis of cyber-criminalistics area, holistic approach to cyber-forensics.

UNIT 2: (8 Lectures)
Cyber Crime Scene Analysis: Discuss the various court orders etc., methods to search and seizure electronic evidence, retrieved and un-retrieved communications, Discuss the importance of understanding what court documents would be required for a criminal investigation.

UNIT 3: (9 Lectures)
Evidence Management & Presentation: Create and manage shared folders using operating system, importance of the forensic mindset, define the workload of law enforcement, Explain what the normal case would look like, Define who should be notified of a crime, parts of gathering evidence, Define and apply probable cause.

UNIT 4: (10 Lectures)
Computer Forensics: Prepare a case, Begin an investigation, Understand computer forensics workstations and software, Conduct an investigation, Complete a case, Critique a case, Network Forensics: open-source security tools for network forensic analysis, requirements for preservation of network data.

UNIT 5: (8 Lectures)
Mobile Forensics: mobile forensics techniques, mobile forensics tools. Legal Aspects of Digital Forensics: IT Act 2000, amendment of IT Act 2008.

UNIT 6: (4 Lectures)
Recent trends in mobile forensic technique and methods to search and seizure electronic evidence.

Suggested Readings/Books:

1. John Sammons, The Basics of Digital Forensics, Elsevier, 2014
2. John Vacca, Computer Forensics: Computer Crime Scene Investigation, Laxmi Publications, 2015

Course title	**Audit Course II: Value Education**
Course Code:	**25C1CCP – 06**
Scheme and Credits	**L T P C** **Semester – II** **2 0 0 0**
Pre-requisite (if any)	
Internal Marks	**0**
External Marks	**0**
Total Marks	**0**

Course Outcomes:

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

CO1	explain the importance of social values, individual attitudes, and work ethics in personal and professional life.
CO2	apply key human values such as honesty, truthfulness, patriotism, and discipline through real-life scenarios and reflective practices.
CO3	analyze personal behavior and personality traits to cultivate a positive mindset and eliminate self-destructive habits.
CO4	evaluate different moral and non-moral values using philosophical and ethical standards to develop a balanced worldview.
CO5	develop a value-based lifestyle by integrating principles from various religions, self-management techniques, and ethical teachings.

Detailed contents:

UNIT 1: (4 Lectures)
Values and self-development –Social values and individual attitudes. Work ethics, Indian vision of humanism. Moral and non- moral valuation. Standards and principles. Value judgements.

UNIT 2: (6 Lectures)
Importance of cultivation of values. Sense of duty. Devotion, Self-reliance. Confidence, Concentration. Truthfulness, Cleanliness. Honesty, Humanity. Power of faith, National Unity. Patriotism, Love for nature, Discipline.

UNIT 3: (6 Lectures)
Personality and Behavior Development - Soul and Scientific attitude. Positive Thinking. Integrity and discipline. Punctuality, Love and Kindness. Avoid fault Thinking. Free from anger, Dignity of labour. Universal brotherhood and religious tolerance. True friendship. Happiness Vs suffering, love for truth. Aware of self-destructive habits. Association and Cooperation. Doing best for saving nature.

UNIT 4: (6 Lectures)

Character and Competence –Holy books vs Blind faith. Self-management and Good health. Science of reincarnation. Equality, Nonviolence, Humility, Role of Women. All religions and same message. Mind your Mind, Self-control. Honesty, Studying effectively.

Suggested Readings/Books:

1. Chakroborty, S.K. “Values and Ethics for organizations Theory and practice”, Oxford University Press, New Delhi, 2024.

Course title	**Mini Project With Seminar**
Course Code:	**25C1CSP-203**
Scheme and Credits	**L T P C** **Semester – II** **2 0 0 2**
Pre-requisite (if any)	
Internal Marks	**60**
External Marks	**40**
Total Marks	**100**

Course Outcomes:

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

CO1	apply advanced programming tools and techniques to design and develop effective software solutions.
CO2	analyze real-world problems and evaluate appropriate methodologies to develop innovative and efficient solutions.
CO3	create structured technical reports that effectively document project work using appropriate writing standards.
CO4	develop ethical behavior, responsibility, and teamwork skills in executing and presenting the mini project.
CO5	build engaging presentations and communicate technical content creatively using effective visual and verbal strategies.

Guidelines:

- Student has to develop quality project on any research topic based on his or her dissertation.
- Any technique need to be implemented.
- The student will have to demonstrate as well as have to give a seminars on the topics related to their research topic.