CENTRAL UNIVERSITY OF PUNJAB, BATHINDA

M. Tech. Computer Science & Technology

Session - 2019-21

Department of Computer Science & Technology

SEMESTER-I

Course Code	Course Title	Course Type	Credit Hours		
			L	**T**	**P**
CST.506	Advanced Data Structure	Core-I	4	0	0
CST.507	Mathematical Foundation of Computer Science	Core-II	4	0	0
CST.508	Machine Learning	Discipline Elective-I	4	0	0
CST.509	Wireless Sensors Networks				
CST.510	Compiler for HPC				
CST.511	Distributed Database System	Discipline Elective-II	4	0	0
CST.512	Information Security				
CST.513	Software Testing & Maintenance				
CST.514	Research Methodology	Compulsory Foundation	4	0	0
XX.YYY	Opt any one course from the courses offered by the University	IDC	2	0	0
CST.515	Advanced Data Structure - Lab	Laboratory-I	0	0	2
CST.516	Wireless Sensors Networks Lab	Laboratory-II	0	0	2
CST.517	Machine Learning Lab				
CST.518	Compiler for HPC- Lab				
Total Credits			**22**	**0**	**4**

SEMESTER-II

Course Code	Course Title	Course Type	Credit Hours		
			L	**T**	**P**
CST.521	Advance Algorithm	Core-III	4	0	0
CST.522	Soft Computing	Core-IV	4	0	0
CST.523	Computer Vision	Discipline Elective-III	4	0	0
CBS.524	Big Data Analysis and Visualization				
CBS.523	Secure Software Design				
CST.524	IOT (Internet of Things)				
CBS.525	Secure Coding	Discipline Elective-IV	4	0	0
CST.525	GPU Computing				
CST.529	Blockchain Technology				
CBS.527	Digital Forensics				
CST.526	Python Programming for Data Sciences	Compulsory Skill Development	4	0	0
XXX.YYY	Inter Disciplinary Course (IDC)	Audit Course	2	0	0

CST.527	Soft Computing-Lab	Laboratory-III	0	0	2
CST.528	Python Programming for Data Science – Lab	Laboratory-IV	0	0	2
Total Credits			**22**	**0**	**4**

SEMESTER-III

Course Code	**Course Title**	**Course Type**	**Credit Hours**		
			L	**T**	**P**
CST.551	Optimization Techniques	Discipline Elective	4	0	0
CST.552	Data Warehousing and Data Mining				
CST.553	Introduction to Intelligent System				
CST.554	Mobile Applications & Services				
CBS.552	Cyber Threat Intelligence	Open Elective	4	0	0
CST.556	Cost Management of Engineering Projects				
CST.557	Software Metrics				
XXX.YYY	Opt any one course from the courses offered by the University	Value Aided	2	0	0
CST.559	Capstone Lab	Core	0	0	2
CST.600	Dissertation/ Industrial Project	Core	0	0	10
Total Credits			**10**	**0**	**12**

*Students going for Industrial Project/Thesis will complete these courses through MOOCs.

SEMESTER-IV

Course Code	**Course Title**	**Course Type**	**Credit Hours**		
			L	**T**	**P**
CST.600	Dissertation	Core	0	0	16
Total Credits			**0**	**0**	**16**

L: Lectures T: Tutorial P: Practical Cr: Credits

SEMESTER – I

L	T	P	Cr
4	0	0	4

Course Code: CST.506
Course Title: Advanced Data Structures
Total Hours: 61

Course Objectives: The objective of this course is to:

- This course will provide the in-depth knowledge of different advance data structures
- Students should be able to understand the necessary mathematical abstraction to solve problems.
- To familiarize students with advanced paradigms and data structure used to solve algorithmic problems.

Learning Outcomes:
After completion of course, students would be able to:

- Discuss various types of data structure and their strengths and weaknesses.
- Compare and contrast non randomized and randomize algorithms
- Develop algorithms for text processing applications.
- Identify suitable data structures and develop algorithms for computational geometry problems.

UNIT I **14 Hours**

Introduction to Basic Data Structures: Importance and need of good data structures and algorithms.

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 II **16 Hours**

Skip Lists: Need for Randomizing Data Structures and Algorithms, Search and Update Operations on Skip Lists, Probabilistic Analysis of Skip Lists, Deterministic Skip Lists.

Binary Search Trees, AVL Trees, Red Black Trees, 2-3 Trees, B-Trees, Splay Trees.

UNIT III **16 Hours**

String Operations, 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 IV **15 Hours**

Computational Geometry: One Dimensional Range Searching, Two Dimensional Range Searching, Constructing a Priority Search Tree, Searching a Priority Search Tree, Priority Range Trees, Quad trees, k-D Trees.

Recent Trends in Hashing, Trees, and various computational geometry methods for efficiently solving the new evolving problem.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. T.H. Cormen, C. E. Leiserson, RL Rivest and C Stein, Introduction to Algorithms, MIT Press.
2. Sridhar, S., Design and Analysis of Algorithms. Oxford University Press India.
3. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, Pearson.
4. M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley.
5. Aho, A.V., Hopcroft, J.E. and Ullman, J. D., Data Structures and Algorithms. India: Pearson Education.
6. Horowitz, E., Sahni, S. and Rajasekaran, S., Fundamentals of Computer Algorithms, Galgotia Publications.

Course Code: CST.507
Course Title: Mathematical Foundation of Computer Science
Total Hours: 64

L	**T**	**P**	**Cr**
4	0	0	4

Course Objectives:

- To understand the mathematical fundamentals that is prerequisites for a variety of courses like Data mining, Network protocols, analysis of Web traffic, Computer security, Software engineering, Bioinformatics, Machine learning.
- To develop the understanding of the mathematical and logical basis to many modern techniques in information technology like machine learning, programming language design, and concurrency.
- To study various sampling and classification problems.

Learning Outcomes:

After completion of course, students would be able to:

- Identify the basic notions of discrete and continuous probability.

- Demonstrate the methods of statistical inference, and the role that sampling distributions play in those methods.
- Design correct and meaningful statistical analyses of simple to moderate complexity problems.
- Synthesize the domain specific mathematical models for different analysis.

UNIT I **17 Hours**

Distribution Function: Probability mass, density, and cumulative distribution functions, Conditional Probability, Expected value,
Applications of the Univariate and Multivariate problems. Probabilistic inequalities, Random samples, sampling distributions of estimators and Maximum Likelihood.

UNIT II **15 Hours**

Statistical inference: Descriptive Statistics, Introduction to multivariate statistical models, Multivariate Regression, Multinominal regression and classification problems, principal components analysis. The problem of overfitting model assessment.
Introduction to Fuzzy Set Theory.

UNIT III **16 Hours**

Graph Theory: Isomorphism, Planar graphs, graph colouring, Hamilton circuits and eulercycles.
Specialized techniques to solve combinatorial enumeration problems Graph Theory: Isomorphism, Planar graphs, graph colouring, Hamilton circuits and Euler cycles. Specialized techniques to solve combinatorial enumeration problems.

UNIT IV **16 Hours**

Computer science and engineering applications with any of following area: Data mining, Computer security, Software engineering, Computer architecture, Bioinformatics, Machine learning.
Recent Trends in various distribution functions in mathematical field of computer science for varying fields like, soft computing, and computer vision.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. John Vince, Foundation Mathematics for Computer Science, Springer International Publishing.

2. Kishor S. Trivedi, Probability and Statistics with Reliability, Queuing, and Computer Science Applications. Wiley.
3. Michel Mitzenmacher and E. Upfal. Probability and Computing: Randomized Algorithms and Probabilistic Analysis, Cambrige University Press.
4. Alan Tucker, Applied Combinatorics, Wiley.

Course Code: CST.508
Course Title: Machine Learning
Total Hours: 63

L	T	P	Cr
4	0	0	4

Course Objectives:

- To explain the concept of how to learn patterns and concepts from data without being explicitly programmed in various IOT nodes.
- To design and analyze various machine learning algorithms and techniques with a modern outlook focusing on recent advances.
- Explore supervised and unsupervised learning paradigms of machine learning.
- To explore Deep learning technique and various feature extraction strategies.

Learning Outcomes:

After completion of course, students would be able to:

- Extract features that can be used for a particular machine learning approach in various IOT applications.
- To compare and contrast pros and cons of various machine learning techniques and to get an insight of when to apply a particular machine learning approach.
- To mathematically analyze various machine learning approaches and paradigms.

UNIT I **16 Hours**

Introduction to learning Techniques: Supervised Learning (Regression/Classification)

- 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 II **15 Hours**

Unsupervised Learning

- Clustering: K-means/Kernel K-means
- Dimensionality Reduction: PCA and kernel PCA
- Matrix Factorization and Matrix Completion

- Generative Models (mixture models and latent factor models)

UNIT III **14 Hours**

Evaluating Machine Learning algorithms and Model Selection, Introduction to Statistical Learning Theory, Ensemble Methods (Boosting, Bagging, Random Forests).
Sparse Modeling and Estimation, Modeling Sequence/Time-Series Data, Deep Learning and Feature Representation Learning.
Introduction to ANN and Deep learning.

UNIT IV **18 Hours**

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.
Simulation Tool for Machine Learning, Hands on with recent tools WEKA, R, MATLAB
Recent trends in various learning techniques of machine learning and classification methods for IOT applications. Various models for IOT applications.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Kevin Murphy, Machine Learning: A Probabilistic Perspective, MIT Press.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Springer.
3. Christopher Bishop, Pattern Recognition and Machine Learning, Springer.
4. Shai Shalev-Shwartz, Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press.

Code: CST. 509
Course Title: Wireless Sensors Networks
Total Hours: 59

L	T	P	Cr
4	0	0	4

Course Objectives:

The objective of this course is to introduce students to:

- The concepts of wireless sensor networks
- Describe various MAC and routing protocols
- Discuss the security for possible attacks

Learning Outcomes:

After completion of course, students would be able to:

- Classify various MAC and routing protocols
- Design wireless sensor networks in simulator.
- Evaluate the performance of various protocols using simulator.

UNIT I **15 Hours**

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.

UNIT II **14 Hours**

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 III **15 Hours**

Routing protocols for WSN: Resource-aware routing, Data-centric, Geographic Routing, Broadcast, Multicast

Opportunistic Routing Analysis: Analysis of opportunistic routing (Markov Chain).

UNIT IV **15 Hours**

Security: Possible attacks, countermeasures, SPINS, Static and dynamic key Distribution.

Introduction to Network Simulations: Introduction to Network Simulator, Description of the module and simulation example.

Advanced Topics: Recent development in WSN standards, software applications.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. W. Dargie and C. Poellabauer, Fundamentals of Wireless Sensor Networks –Theory and Practice, Wiley.
2. Kazem Sohraby, Daniel Minoli and TaiebZnati, wireless sensor networks -Technology, Protocols, and Applications, Wiley.

3. Takahiro Hara,Vladimir I. Zadorozhny, and Erik Buchmann, Wireless Sensor Network Technologies for the Information Explosion Era, springer.
4. Murthy, C.S. R. and Manoj, B.S. Ad hoc Wireless Networks Architectures and protocols, Pearson Education.
5. Obaidat M. S.and Misra, S. Principles of Wireless Sensor Networks, Cambridge University Press, U.K.
6. Misra,S., Woungang, I. and Misra S. C. Guide to Wireless Sensor Networks, Computer Communications and Networks Series, Springer-Verlag, London, U.K.

Code: CST. 510
Course Title: Compiler for HPC
Total Hours: 63

L	T	P	Cr
4	0	0	4

Course Objectives:

- To introduce the structure of compilers and high performance compiler design for students. Concepts of cache coherence and parallel loops in compilers are included.

Learning Outcomes:

After completion of course, students would be able to:

- Familiar with the structure of compiler.
- Discuss regarding parallel loops, data dependency and exception handling and debugging in compiler.

UNIT I **16 Hours**

High Performance Systems, Structure of a Compiler, Programming Language Features, Languages for High Performance.

Data Dependence: Data Dependence in Loops, Data Dependence in Conditionals, Data Dependence in Parallel Loops, Program Dependence Graph.

Scalar Analysis with Factored Use-Def Chains: Constructing Factored Use-Def Chains, FUD Chains for Arrays, Induction Variables Using FUD Chains, Constant Propagation with FUD Chains, Data Dependence for Scalars. Data Dependence Analysis for Arrays.

UNIT II **16 Hours**

Array Region Analysis, Pointer Analysis, I/O Dependence, Procedure Calls, Inter-procedural Analysis.

Loop Restructuring: Simple Transformations, Loop Fusion, Loop Fission, Loop Reversal, Loop Interchanging, Loop Skewing, Linear Loop. Transformations, Strip-Mining, Loop Tiling, Other Loop Transformations, and Inter-Procedural Transformations.

Optimizing for Locality: Single Reference to Each Array, Multiple References, General Tiling, Fission and Fusion for Locality.

UNIT III **14 Hours**

Concurrency Analysis: Concurrency from Sequential Loops, Concurrency from Parallel Loops, Nested Loops, Round off Error, Exceptions and Debuggers.

Vector Analysis: Vector Code, Vector Code from Sequential Loops, Vector Code from for all Loops, Nested Loops, Round off Error, Exceptions, and Debuggers, Multi-vector Computers.

UNIT IV **17 Hours**

Message-Passing Machines: SIMD Machines, MIMD Machines, Data Layout, Parallel Code for Array Assignment, Remote Data Access, Automatic Data Layout, Multiple Array Assignments, Other Topics.

Scalable Shared-Memory Machines: Global Cache Coherence, Local Cache Coherence, Latency Tolerant Machines.

Recent trends in compiler design for high performance computing and message passing machines and scalable shared memory machine.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Michael Wolfe, High-Performance Compilers for Parallel Computing, Pearson.
2. Steven Muchnick, Advanced Compiler Design and Implementation.
3. Allen Optimizing Compilers for Modern Architectures.

Code: CST. 511
Course Title: Distributed Database System
Total Hours: 62

L	T	P	Cr
4	0	0	4

Course Objectives:

- To introduce the fundamental concepts and issues of managing large volume of shared data in a parallel and distributed environment.
- Provide insight into related research problems.

Learning Outcomes:

After completion of course, students would be able to:

- Design trends in distributed systems.
- Apply network virtualization.
- Apply remote method invocation and objects.

UNIT I **14 Hours**

Introduction: 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 II **15 Hours**

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 III **17 Hours**

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 IV **16 Hours**

Reliability: Reliability issues in DDBSs; Types of failures; Reliability techniques; Commit Protocols; Recovery protocols.

Parallel Database Systems: Parallel architectures; parallel query processing and optimization; load balancing.

Introduction to cloud computing, Advanced Topics: Mobile Databases, Distributed Object Management, Multi-databases.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

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

3. Sachin Deshpande, Distributed Databases. Dreamtech Press.
4. Saeed K Rahimi Frank S Haug, Distributed Database Management Systems: A Practical Approach.

Code: CST. 512
Course Title: Information Security
Total Hours: 61

L	T	P	Cr
4	0	0	4

Course Objectives:

- To introduce students to the concept of security, types of attack.
- Help students to understand Symmetric & Asymmetric Key Cryptography
- Give exposure on Internet Security Protocol.

Learning Outcomes:

After completion of course, students would be able to:

- Identify the domain specific security issues.
- Apply Symmetric & Asymmetric Key Cryptography in various applications.
- Analyze Internet Security Protocol.

UNIT I **13 Hours**

History of Information Systems: Importance of Information Systems, Basics of Information Systems, Changing Nature of Information Systems, Global Information Systems.

Essential Security Terminologies: Hardware, Software, Defining Security, Need for Security, Cyber-Crimes, Three Pillars of Security, Introduction to error detection and correction.

UNIT II **16 Hours**

Encryption and Decryption: Attackers and Types of Threats, Encryption Techniques, Classical Cryptographic Algorithms: Monoalphabetic Substitutions such as the Caesar Cipher, Cryptanalysis of Monoalphabetic ciphers, Polyalphabetic Ciphers such as Vigenere, Vernam Cipher, Stream and Block Ciphers.

Symmetric Key Systems: Data encryption Standard (DES), DES Structure, DES Analysis, Multiple DES, Advance Encryption Standard (AES).

UNIT III **16 Hours**

Key Management Protocols: Solving Key Distribution Problem, Diffie-Hellman Key Exchange Algorithm.

Public Key Encryption Systems: Concept and Characteristics of Public Key Encryption System, Rivest-Shamir-Adleman (RSA) Encryption.

Hash Algorithms: Hash concept, Description of Hash Algorithms (MD5 and SHA-1), Digital Signature/Certificate.

UNIT IV **16 Hours**

Internet Security Protocol: Introduction, Secure Socket Layer, Transport Layer Security, Secure Electronic Transaction, 3-D Secure Protocol, Electronic Money, Email Security, Wireless Application Protocol (WAP) Security, Wired Equivalent Privacy (WEP).

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Forouzan, B.A., Cryptography & Network Security. Tata McGraw-Hill Education.
2. Kahate, A., Cryptography and Network Security. McGraw-Hill Higher Ed.
3. Godbole, N., Information Systems Security: Security Management, Metrics, frameworks and Best Practices. John Wiley & Sons India.
4. Stallings, W., Network Security Essentials: applications and standards. Pearson Education India.
5. Stallings, W., Cryptography and Network Security: Principles and Practice, Pearson.
6. Kim, D., and Solomon, M. G., Fundamentals of Information Systems Security. Jones & Bartlett Learning

Code: CST. 513
Course Title: Software Testing & Maintenance
Total Hours: 62

L	T	P	Cr
4	0	0	4

Course Objectives:

- To enable a clear understanding and knowledge of the foundations, techniques, and tools in the area of software testing and its practice in the industry.
- To identify the software testing process for software quality checking.
- To design metrics models for predicting software testing and maintenance requirements.

Learning Outcomes:

After completion of course, students would be able to:

- Apply software testing knowledge, verification & validation and engineering methods.
- Design and conduct a software test process for a quality software test.
- Identify various software testing problems, and solve these problems by designing and selecting software metrics models, testing criteria, strategies, and methods.

UNIT I **14 Hours**

Overview of Software Engineering: Phases in development of Software, Software Engineering Ethics, Life cycle Revisited (Incremental Development, Agile Methods, RAD), Model-Driven Architecture, Software Product Line, Process Modelling**.**

Project Management: Project Planning, Project Control (Work Break Down Structure, GANTT Charts, PERT Charts) Project Team Organization, Risk Management, CMM.

UNIT II **15 Hours**

Testing of OO systems: Objects and Classes, OO Testing, Class Testing, Regression Testing, Non-Functional Testing, Acceptance Testing, Mutation Testing.

Software Testing: Levels of testing, Module, Integration, System, Regression, Testing techniques and their Applicability, Functional testing and Analysis Structural testing and Analysis, Error Oriented testing and Analysis, Hybrid Approaches, Integration Strategies, Transaction Flow Analysis, Stress Analysis, Failure Analysis, Concurrency Analysis.

UNIT III **17 Hours**

Overview of Software Metrics: Measurement in Software Engineering, Scope of Software Metrics, Measurement and Models Meaningfulness in measurement, Measurement quality, Measurement process, Scale, Measurement validation, Object-oriented measurements.

Measuring Internal External Product Attributes: Measuring size, aspects of software size, length, functionality and complexity, measuring structure, types of structural measures, Modeling software quality, measuring aspects of software quality, software reliability, basics of software reliability.

UNIT IV **16 Hours**

Software Maintenance: Maintenance Categories, Major causes of Maintenance Problems, Reverse Engineering, Software Evolutions, Organizational and Managerial Issues of Maintenance activities, Maintenance Measurements Software Refactoring: Principles of Refactoring, Bad Smells in code, Composing Methods of Refactoring, Moving features between objects.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Pressman Roger S. Software Engineering a Practitioners Approach, Latest Edition, McGraw-Hill Singapore.

2. Peters, James S. Witold Pedrycz, Software engineering an engineering approach, Wiley India.
3. Anirban Basu, Software Quality Assurance, Testing and Metrics, Latest Edition, PHI India.
4. Vliet Hans Van, Software Engineering Principles and Practice, John Wiley & Sons.
5. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandriolo, Fundamental of Software Engineering, Latest Edition, Printice Hall India.

Code: CST. 514
Course Title: Research Methodology
Total Hours: 59

L	T	P	Cr
4	0	0	4

Course Objectives:

- To develop a research orientation among the students and help them understand fundamentals of research methods.
- The course will help the students to identify various sources of information for literature review, data collection and effective paper/ dissertation writing.
- Familiarize students with the concept of patents and copyright

Learning Outcomes:

After completion of course, students would be able to:

- Enable the students to effectively formulate a research problem.
- Analyze research related information and follow research ethics.
- Apply intellectual property law principles (including copyright, patents, designs and trademarks) to practical problems and be able to analyse the social impact of IPR.

UNIT I **14 Hours**

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 II **15 Hours**

Effective literature studies approaches, analysis Plagiarism, Research ethics, 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 III **14 Hours**

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 IV **16 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, Integrated Circuits, etc.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Stuart Melville and Wayne Goddard, Research methodology: an introduction for science & engineering students, Juta Academic.
2. Wayne Goddard and Stuart Melville, Research Methodology: An Introduction, Juta Academic.
3. Ranjit Kumar, Research Methodology: A Step by Step Guide for beginners, SAGE Publications Ltd.
4. Halbert, Resisting Intellectual Property, Taylor & Francis Ltd.
5. Mayall , Industrial Design, McGraw Hill.
6. Niebel , Product Design, McGraw Hill.
7. Asimov, Introduction to Design, Prentice Hall.
8. Robert P. Merges, Peter S. Menell, Mark A. Lemley, Intellectual Property in New Technological Age.

Code: CST. 515
Course Title: Advanced Data Structure – Lab

L	T	P	Cr
0	0	4	2

Course Objectives:

- To apply the concept of Dynamic memory management, Data Types.
- To demonstrate the basic data structure such as array, Link List Stack and Queue
- To describe the hash function and concepts of collision and its resolution methods
- To illustrate the problem involving graphs, trees and heaps.
- To Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data.

Learning Outcomes:

After completion of course, students would be able to:

- Implement the basic data structures such as arrays, linked lists, stacks and queues.

- Describe and implement the hash function and concepts of collision and its resolution methods.
- Implement some problem involving graphs, trees and heaps.

Lab Assignments will be based on topics studied in CST-507

Lab Evaluation:
The criteria for evaluation of lab will be based on following parameters:

Component	**Marks**
Continuous Evaluation	30
End Term (Implementation and Viva-Voce)	20
Total	**50**

Code: CST. 516
Course Title: Wireless Sensors Networks Lab

L	**T**	**P**	**Cr**
0	0	4	2

Course Objectives:
The objective of this course is to introduce students to:

- The difference between wired and wireless networks.
- Help them to differentiate between various protocols.
- Describe the various security loopholes and their countermeasures in wireless sensor networks.

Course Outcomes:
After completion of course, students would be able to:

- Design the Wired and Wireless networks using suitable tools.
- Evaluate the performance of sensor networks
- Analyze the wireless sensor networks using various protocols.

List of Practical will be based on Elective – I subject opted by the students

Lab Evaluation:
The criteria for evaluation of lab will be based on following parameters:

Component	**Marks**
Continuous Evaluation	30
End Term (Implementation and Viva-Voce)	20
Total	**50**

Code: CST.517
Course Title: Machine Learning Lab

L	T	P	Cr
0	0	4	2

Course Objectives:

- The objectives of the Machine Learning Lab course are to introduce students to the basic concepts and techniques of Machine Learning.
- To develop skills of using recent machine learning software for solving practical problems.

Learning Outcomes:

After completion of course, students would be able to:

- Understand some common Machine Learning algorithms and their limitations.
- Apply common Machine Learning algorithms in practice and implementing the same.
- Perform experiments in Machine Learning using real-world data.

List of Practical will be based on Elective – I subject opted by the students

Lab Evaluation:

The criteria for evaluation of lab will be based on following parameters:

Component	**Marks**
Continuous Evaluation	30
End Term (Implementation and Viva-Voce)	20
Total	**50**

Code: CST. 518
Course Title: Complier for HPC Lab

L	T	P	Cr
0	0	4	2

Course Objective:

The course is designed to help students understand the

- Concepts like instruction level, data level and thread level parallelism.
- Memory hierarchy optimizations

Course Outcomes:

After the completion of the course the students will be able to

- Learn about some common machine independent optimizations.
- Apply Compiler techniques and tools for exploiting instruction, data and thread level parallelism.
- Gain understanding regarding memory locality optimizations.

List of Practical will be based on Elective – I subject opted by the students

Lab Evaluation:
The criteria for evaluation of lab will be based on following parameters:

Component	**Marks**
Continuous Evaluation	30
End Term (Implementation and Viva-Voce)	20
Total	**50**

SEMESTER -II

L	T	P	Cr
4	0	0	4

Code: CST. 521
Course Title: Advance Algorithm
Total Hours: 61

Course Objectives:

- To familiarize students with basic paradigms and data structures used to solve advanced algorithmic problems
- To introduce the students to recent developments in the area of algorithmic design.

Learning Outcomes:
After completion of course, students would be able to:

- Analyze the complexity/performance of different algorithms.
- Determine the appropriate data structure for solving a particular set of problems.
- Categorize the different problems in various classes according to their complexity.

UNIT I **16 Hours**
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,
Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST.

UNIT II **14 Hours**
Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations.
Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming.

UNIT III **15 Hours**
Linear Programming: Geometry of the feasibility region and Simplex algorithm, Decision Poblems: P, NP, NP Complete, NP-Hard,

NP Hard with Examples, Proof of NP-hardness and NP-completeness.

UNIT IV **16 Hours**

One or more of the following topics based on time and interest Approximation algorithms, Randomized Algorithms, Interior Point Method, Recent Trends in problem solving paradigms using recent searching and sorting techniques by applying recently proposed data structures.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest Introduction to Algorithms, Stein.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, The Design and Analysis of Computer Algorithms.
3. Jon Kleinberg and Eva Tardos , Algorithm Design.
4. Juraj Hromkovic, Design and Analysis of Randomized Algorithms: Introduction to Design.

Code: CST. 522
Course Title: Soft Computing
Total Hours: 58

L	T	P	Cr
4	0	0	4

Course Objectives:

- To introduce soft computing concepts and techniques and foster their abilities in designing appropriate technique for a given scenario.
- To give students knowledge of non-traditional technologies and fundamentals of artificial neural networks, fuzzy sets, fuzzy logic, genetic algorithms.
- To provide student hand-on experience to implement various strategies.

Learning Outcomes:

After completion of course, students would be able to:

- Identify and describe soft computing techniques and their roles in building intelligent machines.
- Apply fuzzy logic and reasoning to handle uncertainty and solve various engineering problems.
- Apply genetic algorithms to combinatorial optimization problems.
- Evaluate and compare solutions by various soft computing approaches for a given problem.

UNIT I **14 Hours**

Introduction to Soft Computing and Neural Networks: Evolution of Computing: Soft Computing Constituents, From Conventional AI to Computational Intelligence: Machine Learning Basics. Adaptive Resonance architectures, Advances in Neural networks.

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.

UNIT II **14 Hours**

Fuzzy Logic: Fuzzy Sets, Membership Functions, Operations on Fuzzy Sets, Fuzzy Relations.

Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making, Fuzzy Models.

UNIT III **16 Hours**

Genetic Algorithms: Introduction to Genetic Algorithms (GA), Applications of GA in Machine Learning: Machine Learning Approach to Knowledge Acquisition. Introduction to other optimization techniques.

UNIT IV **14 Hours**

Implementation of simple artificial neural networks, fuzzy logic techniques and genetic algorithms.

Recent trends in soft computing techniques. Introduction to hybrid systems and swarm intelligence.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro - Fuzzy and Soft Computing, Prentice-Hall of India.
2. George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic - Theory and Applications, Prentice Hall.
3. Ross J.T., Fuzzy Logic with Engineering Applications John Wiley & Sons.
4. Rajasekaran, S. Vijayalakshmi Pai, G.A. Neural Networks, Fuzzy Logic and Genetic Algorithms PHI Learning.
5. Priddy L.K., Keller E.P., Artificial Neural Networks: An Introduction, SPIE Press.
6. Gen, M. Cheng R., Genetic Algorithms and Engineering Optimization John Wiley & Sons.

Code: CST. 523
Course Title: Computer Vision

L	T	P	Cr
4	0	0	4

Total Hours: 61
Course Objectives:

- To review both the theoretical and practical aspects of computing with images for computer vision.
- To be able to understand image formation, measurements, and analysis.
- Describe the geometric relationships between 2D images and the 3D world.
- Grasp the principles of state-of-the-art feature extraction and classification techniques.

Learning Outcomes:
After completion of course, students would be able to:

- Describe the various image processing and analysis methods for computer vision.
- Compare and contrast various object and scene recognition, classification and clustering techniques.
- Developed the practical skills necessary to build computer vision applications.

UNIT I **15 Hours**
Overview, computer imaging systems, lenses, Image formation and sensing, Image analysis, pre-processing and binary image analysis.
Edge detection, Edge detection performance, Hough transform, corner detection

UNIT II **16 Hours**
Segmentation, Morphological filtering, Fourier transform Feature extraction, shape, histogram, color, spectral, texture, using CVIP tools.

UNIT III **16 Hours**
Feature analysis, feature vectors, distance /similarity measures, data preprocessing
Pattern Analysis: Clustering: K-Means, K-Medoids, Mixture of Gaussians
Classification: Discriminant Function, Supervised, Un-supervised, Semi supervised

UNIT IV **14 Hours**
Classifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA, ICA, and Non-parametric methods.
Recent trends in Activity Recognition, computational photography, Biometrics.

Transactional Modes:
Lecture
Case study

Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Richard Szeliski, Computer Vision - Algorithms and Applications, Springer.
2. Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning, MIT Press.
3. Robert B. Fisher, Kenneth Dawson-Howe, Andrew Fitzgibbon, Dictionary of Computer Vision and Image Processing, Wiley.
4. Reinhard Klette, Concise Computer Vision: An Introduction into Theory and Algorithms,.
5. Earl Gose, Richard Johnsonbaugh, Steve, Pattern Recognition and Image Analysis.

Code: CBS.524
Course Title: Big Data Analysis and Visualization
Total Hours: 61

L	T	P	Cr
4	0	0	4

Course Objectives:

- To prepare the Big Data for analysis.
- To extract the meaningful data from unstructured Big Data and develop Data Visualizations skill.
- To apply various tools for analysis of structured and unstructured Big Data.

Learning Outcomes:
After completion of course, students would be able to:

- Able to analyse the identification of Big Data problem
- Able to extract the structured data from unstructured data.
- Use Hadoop related tools such as JAQL, Spark, Pig and Hive for structured and unstructured Big Data analytics

UNIT I **15 Hours**
Big Data Introduction: What is big data, why big data, convergence of key trends, unstructured data, industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data, big data and healthcare, big data in medicine, advertising and big data, big data technologies, open source technologies, cloud and big data, mobile business intelligence, Crowd sourcing analytics, inter and trans firewall analytics.
Data Gathering and Preparation: Data formats, parsing and transformation, Scalability and real-time issues.

UNIT II **16 Hours**
Data Cleaning: Consistency checking, Heterogeneous and missing data, Data Transformation and segmentation.

Visualization: Descriptive and comparative statistics, Designing visualizations, Time series, Geo-located data, Correlations and connections, Hierarchies and networks, interactivity.

UNIT III **15 Hours**

Big Data Technology: Big Data Architecture, Big Data Warehouse,
Functional Vs. Procedural Programming Models for Big Data
NoSQL: Introduction to NoSQL, aggregate data models, key-value and document data models.

UNIT IV **15 Hours**

Big Data Tools: Hadoop: Introduction to Hadoop Ecosystem, HDFS, Map-Reduce programming, Spark, PIG, JAQL, Understanding Text Analytics and Big Data, Predictive Analysis of Big Data, Role of Data Analyst.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, John WIley & Sons.
2. Anil Maheshwari, Data Analytics Make Accesible, Orilley Publications.
3. Croll and B. Yoskovitz Lean Analytics: Use Data to Build a Better Startup Faster, Oreilley Publications.

L	T	P	Cr
4	0	0	4

Code: CBS.523
Course Title: Secure Software Design
Total Hours: 61

Course Objectives:

- To fix software flaws and bugs in various software.
- To make students aware of various issues like weak random number generation, information leakage, poor usability, and weak or no encryption on data traffic.
- Techniques for successfully implementing and supporting network services on an enterprise scale and heterogeneous systems environment.
- Methodologies and tools to design and develop secure software containing minimum vulnerabilities and flaws.

Learning Outcomes:

After completion of course, students would be able to:

- Differentiate between various software vulnerabilities.
- Software process vulnerabilities for an organization.
- Monitor resources consumption in a software.
- Interrelate security and software development process.

UNIT I **13 Hours**

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 II **15 Hours**

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 III **16 Hours**

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 IV **17 Hours**

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.

Handle insecure exceptions and command/SQL injection, Defend web and mobile applications against attackers, software containing minimum vulnerabilities and flaws. Case study of DNS server, DHCP configuration and SQL injection attack.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Theodor Richardson, Charles N Thies, Secure Software Design, Jones & Bartlett Learning.
2. Kenneth R. van Wyk, Mark G. Graff, Dan S. Peters, Diana L. Burley, Enterprise Software Security: A Confluence of Disciplines, Addison - Wesley, Professional.
3. Gary, Software Security: Building Security, McGraw.
4. Dafydd Stuttard, The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws.
5. Jan Erik Solem, Programming Computer Vision with Python: Tools and algorithms for analyzing images.

Code: CST.524
Course Title: IOT (Internet of Things)
Total Hours: 54

L	T	P	Cr
4	0	0	4

Course Objectives:

The objective of this course is to introduce students to

1. Explain the concepts of Internet of Things
2. The use of devices in IoT Technology,
3. Identify Real World IoT Design Constraints.

Learning Outcomes:

After completion of course, students would be able to:

1. Analyze the challenges in IoT Design
2. Describe the domain specific applications
3. Design IoT applications on different embedded platform.

UNIT I **10 Hours**

Introduction to IoT: Defining IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models and APIs IoT and M2M, Difference between IoT and M2M, Software define Network.

UNIT II **12 Hours**

Network and Communication aspects: Wireless medium access issues, MAC protocol survey, Survey routing protocols, Sensor deployment, Node discovery, Data aggregation and Dissemination.

UNIT III **16 Hours**

Challenges in IoT Design: challenges, Development challenges, Security challenges, Other Challenges

Domain specific applications: IoT Home automation, Industry applications, Surveillance applications, Other IoT applications

UNIT IV **16 Hours**

Developing IoTs: Developing applications through IoT tools including Python/Arduino/Raspberry pi, Developing sensor based application through embedded system platform.

Transactional Modes:
Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Vijay Madisetti, Arshdeep Bahga, Internet of Things: A Hands-On Approach, Orient Blackswan Pvt. Ltd.- New Delhi.
2. Waltenegus Dargie, Christian Poellabauer, Fundamentals of Wireless Sensor Networks: Theory and Practice, Wiley-Blackwell.
3. Francis da Costa, Rethinking the Internet of Things: A Scalable Approach to Connecting Everything, Apress Publications.
4. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, Academic Press.

L	**T**	**P**	**Cr**
4	0	0	4

Code: CBS.525
Course Title: Secure Coding
Total Hours: 53

Course Objectives:
The objective of this course is to:

- Explain the most frequent programming errors leading to software vulnerabilities.
- Identify security problems in software.
- Define security threats and software vulnerabilities.

Learning Outcomes:
After completion of course, students would be able to:

- Define secure programs and disk various risk in the softwares.
- Classify various errors that lead to vulnerabilities.
- Analyze various possible security attacks.

UNIT I **11 Hours**

Software Security: Security Concepts, Security Policy, Security Flaws, Vulnerabilities, Exploitation and Mitigations. Software Security problems, Classification of Vulnerabilities.
Security Analysis: Problem Solving with static analysis: Type Checking, Style Checking, Program understanding, verifications and property checking, Bug finding and Security Review.

UNIT II **14 Hours**

Strings: Common String manipulating Errors, String Vulnerabilities and Exploits, Mitigation Strategies for strings, String handling functions, Runtime protecting strategies, Notable Vulnerabilities.

Integer Security: Integer data Type, Integer Conversions, Integer Operations, Integer Vulnerabilities, Mitigation Strategies.

UNIT III **15 Hours**

Handling Inputs: What to validate, How to validate, Preventing metadata Vulnerabilities.

Buffer Overflow: Introduction, Exploiting buffer overflow vulnerabilities, Buffer allocation strategies**,** Tracking buffer sizes, buffer overflow in strings, Buffer overflow in Integers Runtime protections

UNIT IV **13 Hours**

Web Applications: Input and Output Validation for the Web: Expect That the Browser Has Been Subverted, HTTP Considerations: Use POST, Not GET, Request Ordering, Error Handling, Request Provenance

Maintaining Session State: Use Strong Session Identifiers, Enforce a Session Idle Timeout and a Maximum Session Lifetime, Begin a New Session upon Authentication.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Seacord, R. C., Secure Coding in C and C++, Addison Wisley.
2. Chess, B., and West, J., Secure Programming with static Analysis, Addison Wisley.
3. Seacord, R. C., The CERT C Secure Coding Standard, Pearson Education.
4. Howard, M., LeBlanc, D., Writing Secure Code, Pearson Education.

Code: CST.525
Course Title: GPU Computing
Total Hours: 61

L	T	P	Cr
4	0	0	4

Course Objectives:

- To learn parallel programming with Graphics Processing Units (GPUs).

Learning Outcomes:

After completion of course, students would be able to:

- Students would learn concepts in parallel programming, implementation of programs on GPUs, debugging and profiling parallel programs.

UNIT I **13 Hours**

Introduction: History, Graphics Processors, Graphics Processing Units, GPGPUs. Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel programming, CUDA Open CL / Open ACC, Hello World Computation Kernels, Launch parameters, Thread hierarchy, Warps / Wave fronts, Thread blocks / Workgroups, Streaming multiprocessors, 1D / 2D /3D thread mapping, Device properties, Simple Programs.

UNIT II **15 Hours**

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 III **16 Hours**

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 IV **17 Hours**

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.

Case Studies: Image Processing, Graph algorithms, Simulations, Deep Learning

Advanced topics: Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. David Kirk, Wen-mei Hwu, Morgan Kaufman, Programming Massively Parallel Processors: A Hands-on Approach.
2. Shane Cook, Morgan Kaufman, CUDA Programming: A Developer's Guide to Parallel Computing with GPUs.

Code: CST.529
Course Title: Blockchain Technology
Total Hours: 63

L	T	P	Cr
4	0	0	4

Course Objectives:

The objective of this course is to introduce students to:

- Define the concept of Blockchain, Crypto Primitives, Bitcoin Basics.
- Explain distributed Consensus, and Consensus in Bitcoin
- Discuss Permissioned Blockchain, and Hyperledger Fabric.

Learning Outcomes:

After completion of course, students would be able to:

- Describe the basic concept of Blockchain, Crypto Primitives, Bitcoin Basics
- Identify the area in which they can apply permission or permission less blockchain.
- Apply Block chaining concept in various applications.

UNIT I **15 Hours**

Introduction to Blockchain: What is Blockchain, Public Ledgers, Blockchain as Public Ledgers, Bitcoin, Blockchain 2.0, Smart Contracts, Block in a Blockchain, Transactions, Distributed Consensus, The Chain and the Longest Chain, Cryptocurrency to Blockchain 2.0, Permissioned Model of Blockchain

UNIT II **15 Hours**

Basic Crypto Primitives: Cryptographic Hash Function, Properties of a hash function, Hash pointer and Merkle tree, Digital Signature, Public Key Cryptography, A basic cryptocurrency.

Bitcoin Basics: Creation of coins, Payments and double spending, FORTH – the precursor for Bitcoin scripting, Bitcoin Scripts, Bitcoin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation and block relay.

UNIT III **16 Hours**

Distributed Consensus: Why Consensus, Distributed consensus in open environments, Consensus in a Bitcoin network.

Consensus in Bitcoin: Bitcoin Consensus, Proof of Work (PoW) – basic introduction, Hashcash PoW, Bitcoin PoW, Attacks on PoW and the monopoly problem, Proof of Stake, Proof of Burn and Proof of Elapsed Time. The life of a Bitcoin Miner, Mining Difficulty, Mining Pool.

Permissioned Blockchain: Permissioned model and use cases, Design issues for Permissioned blockchains, Execute contracts, State machine replication, Consensus models for permissioned blockchain, Distributed consensus in closed environment, Paxos, RAFT Consensus, Byzantine general problem.

UNIT IV **17 Hours**

Blockchain Components and Concepts: Actors in a Blockchain, Components in Blockchain design, Ledger in Blockchain.

Hyperledger Fabric – Transaction Flow: Fabric Architecture, Transaction flow in Fabric.

Hyperledger Fabric Details: Ordering Services, Channels in Fabric, Fabric Peer and Certificate Authority.

Fabric – Membership and Identity Management: Organization and Consortium Network, Membership Service Provide, Transaction Signing.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Nitin Gaur, Luc Desrosiers, Venkatraman Ramakrishna, Petr Novotny, Salman Baset, Anthony O'Dowd.Hands-On Blockchain with Hyperledger: Building decentralized applications with Hyperledger Fabric and Composer. Packt Publishing Ltd.
2. Bellaj Badr, Richard Horrocks, Xun (Brian) Wu. Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger. Packt Publishing Ltd, 2018.
3. Vikram Dhillon, David Metcalf, Max Hooper. Blockchain Enabled Applications: Understand the Blockchain Ecosystem and How to Make it Work for You. Apress.
4. Mayukh Mukhopadhyay Ethereum Smart Contract Development: Build blockchain-based decentralized applications using solidity. Packt Publishing Ltd.

Code: CBS.527
Course Title: Digital Forensics
Total Hours: 57

L	T	P	Cr
4	0	0	4

Course Objectives:

- Provides an in-depth study of the rapidly changing and fascinating field of computer forensics.
- Combines both the technical expertise and the knowledge required to investigate, detect and prevent digital crimes.
- Knowledge on digital forensics legislations, digital crime, forensics processes and procedures, data acquisition and validation, e-discovery tools.

- E-evidence collection and preservation, investigating operating systems and file systems, network forensics, art of steganography and mobile device forensics.

Learning Outcomes:

After completion of course, students would be able to:

- Explain relevant legislation and codes of ethics.
- Computer forensics and digital detective and various processes, policies and procedures.
- E-discovery, guidelines and standards, E-evidence, tools and environment.
- Email and web forensics and network forensics.

UNIT I **15 Hours**

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.

Legal Aspects of Digital Forensics: IT Act 2000, amendment of IT Act 2008.

UNIT II **14 Hours**

Incident- Response Methodology, 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 III **12 Hours**

Image Capturing, Authenticating Evidence, Hidden Data Extraction, Data Storage, File Systems, Recovery of deleted files, Cracking Passwords,
Internet Crime Investigations, Web Attack Investigations.

UNIT IV **16 Hours**

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.

Mobile Forensics: mobile forensics techniques, mobile forensics tools

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. John Sammons, The Basics of Digital Forensics, Elsevier.
2. Davidoff, S. and Ham, J., Network Forensics Tracking Hackers through Cyberspace, Prentice Hall.
3. Michael G. Solomon, K Rudolph, Ed Tittel, Broom N., and Barrett D., Computer Forensics Jump Start, Willey Publishing, Inc,.
4. Marcella, Albert J., Cyber forensics: A field manual for collecting, examining and preserving evidence of computer crimes, New York, Auerbach publications.
5. Davidoff, Sherri, Network forensics: Tracking hackers through cyberspace, Pearson education India.

Code: CST.526
Course Title: Python Programing for Data Sciences
Total Hours: 63

L	T	P	Cr
4	0	0	4

Course Objectives:

- To introduce students to the Python programming language.
- To handle object oriented problems with Python code.
- To Produce Python code to statistically analyse a dataset

Learning Outcomes:

After completion of course, students would be able to:

- Define python environment and constructs of Python language.
- Construct scripts in Python language.
- Analyse data with Python Libraries.

UNIT I **16 Hours**

Python Introduction: Installing and setting Python environment in Windows and Linux, basics of Python interpreter, Execution of python program, Editor for Python code, syntax, variable, types. Flow control: if, if-else, for, while, range () function, continue, pass, break. Strings: Sequence operations, String Methods, Pattern Matching.

UNIT II **16 Hours**

Lists: Basic Operations, Iteration, Indexing, Slicing and Matrixes; Dictionaries: Basic dictionary operations; Tuples: Basic Operations, Iteration, Indexing, Slicing; Functions: Definition, Call, Arguments, Scope rules and Name resolution;

Modules: Module Coding Basics, Importing Programs as Modules, Executing Modules as Scripts, Compiled Python files(.pyc), Standard Modules: OS and SYS, The dir() Function, Packages.

UNIT III **14 Hours**

Object Oriented Programming in Python: Classes, Objects, Inheritance, Operator Overloading, File Handling: Errors and Exceptions Handling (try and except) User-Defined Exception Objects, Regular expressions, User Defined Package with Python.

UNIT IV **17 Hours**

Python Packages for Data Sciences: Mathematical and Statistical Analysis with NumPy, Manuplating and Visulisation of Data with SciPy, shaping, merging, reshaping, slicing datasets and Data structure with Pandas Library, 2d Plot with matplotlib and seaborn, Learning Package: sklearn

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Mark Lutz., Learning Python, Latest Edition, O'REILLY Media, Inc.
2. Paul Berry. Head First Python, O'REILLY Media, Inc.
3. Jeeva Jose & P. Sojan Lal., Introduction to Computing & Problem Solving with Python.

Code: CST.527
Course Title: Soft Computing Lab

L	T	P	Cr
0	0	4	2

Course Objectives: The primary objective of this lab course is to provide a practical introduction to various techniques in soft computing and their applications.

Course outcome:

After Completion of the lab course the students will be able to:

- Implement simple applications using the fuzzy logic.
- Understand the various types of neural networks and write programmes to implement the same.
- Learn optimization based on GA and implement some of its applications.

Students will implement the lab practical as per the syllabus of the subject.

Lab Evaluation:

The evaluation of lab criteria will be based on following parameters:

Component	**Marks**
Continuous Evaluation	30
End Term (Implementation and Viva-Voce)	20
Total	**50**

Code: CST.528
Course Title: Python Programming for Data Sciences Lab

L	T	P	Cr
0	0	4	2

Course Objectives:

- To understand the basic constructs of Python Interpreter.
- To demonstrate the working of Python functions and modules w.r.t definition call and scope.
- To make acquainted with OOPS and File handling concept in Python.
- To understand and apply various Python packages for Data handling.

Course outcome:

After Completion of the lab course the students will be able to:

- Write and demonstrate script in Python by using basic constructs and control statements of Python.
- Illustrate the use of OOPS and file handling concept for data handling and visualisation.
- Synthesize the code in Python by making a use of various Data Handling libraries.

Students will implement the lab practical as per the syllabus of the subject.

Lab Evaluation:

The evaluation of lab criteria will be based on following parameters:

Component	Marks
Continuous Evaluation	30
End Term (Implementation and Viva-Voce)	20
Total	**50**

SEMESTER-III

Code: CST.551
Course Title: Optimization Techniques

L	T	P	Cr
4	0	0	4

Total Hours: 56

Course Objectives:

- The objective of this course is to provide insight to the mathematical formulation of real world problems.
- To optimize these mathematical problems using nature based algorithms. And the solution is useful especially for NP-Hard problems.

Learning Outcomes:

After completion of course, students would be able to:

- Formulate optimization problems.
- Understand and apply the concept of optimality criteria for various

types of optimization problems.

- Solve various constrained and unconstrained problems in Single variable as well as multivariable.
- Apply the methods of optimization in real life situation.

UNIT I 14 Hours

Engineering application of Optimization, Formulation of design problems as mathematical programming problems.

General Structure of Optimization Algorithms, Constraints, the Feasible Region.

UNIT II 12 Hours

Branches of Mathematical Programming: Optimization using calculus, Graphical Optimization, Linear Programming, Quadratic Programming, Integer Programming, Semi Definite Programming.

UNIT III 16 Hours

Optimization Algorithms like Genetic Optimization, Particle Swarm Optimization, Ant Colony Optimization etc.

UNIT IV 14 Hours

Real life Problems and their mathematical formulation as standard programming problems. Recent trends: Applications of ant colony optimization, genetics and linear and quadratic programming in real world applications.

Transactional Modes:

Lecture

Case study

Demonstration

Experimentation

Discussion

Problem solving

Suggested Readings:

1. Laurence A. Wolsey Integer programming, Wiley.
2. Andreas Antoniou, Wu- Sheng Lu, Practical Optimization Algorithms and Engineering Applications, Springer.
3. Edwin K., P. Chong & Stanislawh. Zak., An Introduction to Optimization, Wiley-Interscience.
4. Dimitris Bertsimas, Robert Weismantel, Optimization over integers Dynamic Ideas.
5. John K. Karlof, Integer programming: theory and practice CRC Press.
6. H. Paul Williams, Logic and Integer Programming Springer.
7. Der-San Chen, Robert G. Batson, Yu Dang, Applied Integer Programming: Modeling and Solution, John Wiley and Sons.

Code: CST.552

Course Title: Data Warehousing and Data Mining
Total Hours: 62

L	T	P	Cr
4	0	0	4

Course Objectives:
The objective of this course is to introduce data warehousing and mining techniques. Application of data mining in web mining, pattern matching and cluster analysis is included to aware students of broad data mining areas.

Learning Outcomes:
After completion of course, students would be able to:

- Study of different sequential pattern algorithms.
- Study the technique to extract patterns from time series data and it application in real world.
- Can extend the Graph mining algorithms to Web mining.
- Help in identifying the computing framework for Big Data.

UNIT I **14 Hours**
Introduction to Data Warehousing: Data warehousing Architecture, OLAP Server, Data warehouse Implementation.
Data Mining: Mining frequent patterns, association and correlations; Sequential Pattern Mining concepts, primitives, scalable methods;

UNIT II **15 Hours**
Classification and prediction: Cluster Analysis – Types of Data in Cluster Analysis, Partitioning methods, Hierarchical Methods; Transactional Patterns and other temporal based frequent patterns.

UNIT III **16 Hours**
Mining Time series Data, Periodicity Analysis for time related sequence data, Trend analysis, Similarity search in Time-series analysis;
Mining Data Streams, Methodologies for stream data processing and stream data systems, Frequent pattern mining in stream data, Sequential Pattern Mining in Data Streams, Classification of dynamic data streams.

UNIT IV **17 Hours**
Web Mining, Mining the web page layout structure, mining web link structure, mining multimedia data on the web, Automatic classification of web documents and web usage mining; Distributed Data Mining.
Recent trends in Distributed Warehousing and Data Mining, Class Imbalance Problem; Graph Mining; Social Network Analysis.

Transactional Modes:
Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Jiawei Han and M Kamber, Data Mining Concepts and Techniques, Elsevier Publication.
2. Vipin Kumar, Michael Steinbach, Introduction to Data Mining - Pang-Ning Tan, Addison Wesley.
3. G Dong and J Pei, Sequence Data Mining, Springer.

Code: CST.553
Course Title: Introduction to Intelligent Systems
Total Hours: 60

L	T	P	Cr
4	0	0	4

Course Objectives:

- The aim of the course is to introduce to the field of Artificial Intelligence (AI) with emphasis on its use to solve real world problems for which solutions are difficult to express using the traditional algorithmic approach.
- It explores the essential theory behind methodologies for developing systems that demonstrate intelligent behaviour including dealing with uncertainty, learning from experience and following problem solving strategies found in nature.

Learning Outcomes:

After completion of course, students would be able to:

- Able to demonstrate knowledge of the fundamental principles of intelligent systems and would be able to analyse and compare the relative merits of a variety of AI problem solving techniques.

UNIT I **15 Hours**

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 II **15 Hours**

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 III **15 Hours**

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 IV **15 Hours**

Biological foundations to intelligent systems I: Artificial neural networks, Back propagation Networks, Radial basis function networks, and recurrent networks.

Biological foundations to intelligent systems II: Fuzzy logic, knowledge Representation and inference mechanism, genetic algorithm, and fuzzy neural networks.

Recent trends in Fuzzy logic, Knowledge Representation

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Luger G.F. and Stubblefield W.A., Artificial Intelligence: Structures and strategies for Complex Problem Solving, Addison Wesley.
2. Russell S. and Norvig P., Artificial Intelligence: A Modern Approach, Prentice-Hall.

Code: CST.554
Course Title: Mobile Applications & Services
Total Hours: 62

L	T	P	Cr
4	0	0	4

Course Objectives:

- This course presents the three main mobile platforms and their ecosystems, namely Android, iOS, and PhoneGap/Web OS.
- It explores emerging technologies and tools used to design and implement feature-rich mobile applications for smartphones and tablets

Learning Outcomes:

After completion of course, students would be able to:

- On completion of the course the student should be able to identify the target platform and users and be able to define and sketch a mobile application.
- Understand the fundamentals, frameworks, and development lifecycle of mobile application platforms including iOS, Android, and PhoneGap.
- Design and develop a mobile application prototype in one of the platform (challenge project).

UNIT I 14 Hours

Introduction: Introduction to Mobile Computing, Introduction to Android Development Environment, Factors in Developing Mobile Applications, Mobile Software Engineering, Frameworks and Tools, Generic UI Development Android User.

UNIT II 15 Hours

More on Uis: VUIs and Mobile Apps, Text-to-Speech Techniques, Designing the Right UI, Multichannel and Multimodal Uis, . Storing and Retrieving Data, Synchronization and Replication of Mobile Data, Getting the Model Right, Android Storing and Retrieving Data, Working with a Content Provider

UNIT III 16 Hours

Communications via Network and the Web: State Machine, Correct Communications Model, Android Networking and Web, Telephony Deciding Scope of an App, Wireless Connectivity and Mobile Apps, Android Telephony Notifications and Alarms-Performance, Performance and Memory Management, Android Notifications and Alarms, Graphics, Performance and Multithreading, Graphics and UI Performance, Android Graphics.

UNIT IV 17 Hours

Putting It All Together: Packaging and Deploying, Performance Best Practices, Android Field Service App, Location Mobility and Location Based Services Android Multimedia: Mobile Agents and Peer-to-Peer Architecture, Android Multimedia Platforms and Additional Issues: Development Process, Architecture, Design, Technology Selection, Mobile App Development Hurdles, Testing, Security and Hacking, Active Transactions, More on Security, Hacking Android.

Recent trends in Communication protocols for IOT nodes, mobile computing techniques in IOT, agents based communications in IOT.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Wei-Meng Lee, Beginning Android TM 4 Application Development, John Wiley & Sons.

Code: CST.557
Course Title: Software Metrics
Total Hours: 58

L	T	P	Cr
4	0	0	4

Course Objectives:

Understand the underlying concepts, principles and practices in Software Measurements. Designing of Metrics model for software quality prediction and reliability.

Learning Outcomes:

After completion of course, students would be able to:

- Able to learn role software Metrics in Industry size software
- Empirical investigation of software for a quality measurement.
- Understand and identify software reliability and problem solving by designing and selecting software reliability models.

UNIT I **14 Hours**

Overview of Software Metrics: Measurement in Software Engineering, Scope of Software Metrics, Measurement and Models Meaningfulness in measurement, Measurement quality, Measurement process, Scale, Measurement validation, Object-oriented measurements.

Goal based framework for software measurement: Software measure classification, Goal-Question-Metrics(GQM) and Goal-Question-Indicator-Metrics (GQIM), Applications of GQM and GQIM.

UNIT II **15 Hours**

Empirical Investigation: Software engineering investigation, Investigation principles, Investigation techniques, Planning Formal experiments, Case Studies for Empirical investigations.

Object–oriented metrics: Object-Oriented measurement concepts, Basic metrics for OO systems, OO analysis and design metrics, Metrics for productivity measurement, Metrics for OO software quality.

UNIT III **16 Hours**

Measuring Internal Product attributes: Software Size, Length, reuse, Functionality, Complexity, Software structural measurement, Control flow structure, Cyclomatic Complexity, Data flow and data structure attributes Architectural measurement.

Measuring External Product attributes: Software Quality Measurements, Aspectes of Quality Measurements, Maintainability Measurements, Usability and Security Measurements.

UNIT IV **13 Hours**

Measuring software Reliability: Concepts and definitions, Software reliability models and metrics, Fundamentals of software reliability engineering (SRE), Reliability management model.

Transactional Modes:

Lecture

Case study

Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Norman E. Fenton, S. L. P fleeger, Software Metrics: A Rigorous and Practical Approach, published by International Thomson Computer Press.
2. Stephen H. Kan, Metrics and Models in Software Quality Engineering, Addison-Wesley Professional.
3. Basu Anirban, Software Quality Assurance, Testing and Metrics, Prentice Hall India.

Code: CBS.552
Course Title: Cyber threat Intelligence
Total Hours: 62

L	T	P	Cr
4	0	0	4

Course Objectives:

The objective of this course is to introduce students to:

- Explain the cyber threats and Cyber Threat Intelligence Requirements
- Classify Cyber Threat Information
- Examine the potential for incidents and, provide more thoughtful responses.

Learning Outcomes:

After completion of course, students would be able to:

- Describe different Cyber Threat.
- Explain technique to Develop Cyber Threat Intelligence Requirements.
- Analyze and Disseminating Cyber Threat Intelligence

UNIT I **15 Hours**

Defining Cyber Threat Intelligence: The Need for Cyber Threat Intelligence: The menace of targeted attacks, The monitor-and-respond strategy, Why the strategy is failing, Cyber Threat Intelligence Defined, Key Characteristics: Adversary based, Risk focused, Process oriented, Tailored for diverse consumers, The Benefits of Cyber Threat Intelligence

UNIT II **14 Hours**

Developing Cyber Threat Intelligence Requirements: Assets That Must Be Prioritized: Personal information, Intellectual property, Confidential business information, Credentials and IT systems information, Operational systems. Adversaries: Cybercriminals, Competitors and cyber espionage agents, Hacktivists. Intelligence Consumers: Tactical users, Operational users, Strategic users

UNIT III **17 Hours**

Collecting Cyber Threat Information: Level 1: Threat Indicators, File hashes and reputation data, Technical sources: honeypots and scanners, Industry sources: malware and reputation feeds. Level 2: Threat Data Feeds, Cyber threat statistics, reports, and surveys, Malware analysis. Level 3: Strategic Cyber Threat Intelligence, Monitoring the underground, Motivation and intentions, Tactics, techniques, and procedures.

Analyzing and Disseminating Cyber Threat Intelligence: Information versus Intelligence, Validation and Prioritization: Risk scores, Tags for context, Human assessment. Interpretation and Analysis: Reports, Analyst skills, Intelligence platform, Customization. Dissemination: Automated feeds and APIs, Searchable knowledge base, Tailored reports.

UNIT IV **16 Hours**

Selecting the Right Cyber Threat Intelligence Partner: Types of Partners: Providers of threat indicators, Providers of threat data feeds, Providers of comprehensive cyber threat intelligence. Important Selection Criteria: Global and cultural reach, Historical data and knowledge, Range of intelligence deliverables, APIs and integrations, Intelligence platform, knowledge base, and portal, Client services, Access to experts. Intelligence-driven Security.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Jon Friedman. Mark Bouchard, CISSP. Foreword by John P. Watters, Cyber Threat Intelligence, Definitive Guide TM.
2. Scott J. Roberts, Rebekah Brown, Intelligence- Driven Incident Response: Outwitting the Adversary, O'Reilly Media.
3. Henry Dalziel, How to Define and Build an Effective Cyber Threat Intelligence Capability Elsevier Science & Technology.
4. John Robertson, Ahmad Diab, Ericsson Marin, Eric Nunes, Vivin Paliath, Jana Shakarian, Paulo Shakarian, DarkWeb Cyber Threat Intelligence Mining Cambridge University Press.
5. Bob Gourley, The Cyber Threat, Createspace Independent Pub.

Code: CST.556
Course Title: Cost Management of Engineering Projects
Total Hours: 55

L	T	P	Cr
4	0	0	4

Course Objectives

- This course provides students with skills and knowledge of cost management of engineering projects.
- The course will enable students to understand the key components of engineering project

Course Outcomes:

- After the completion of the course the students will be able to
- Link their knowledge and skills together to understand the basis of a successful project.
- Understand the cost behaviour and profit planning
- Describe various quantitative methods for cost management

UNIT I **11 Hours**

Introduction and Overview of the Strategic Cost Management Process
Cost concepts in decision-making; Relevant cost, Differential cost, Incremental cost and Opportunity cost. Objectives of a Costing System; Inventory valuation; Creation of a Database for operational control; Provision of data for Decision-Making.

UNIT II **14 Hours**

Project: meaning, Different types, why to manage, cost overruns centers, various stages of project execution: conception to commissioning. Project execution as conglomeration of technical and nontechnical activities. Detailed Engineering activities. Pre project execution main clearances and documents Project team: Role of each member. Importance Project site: Data required with significance.
Project contracts. Types and contents. Project execution Project cost control. Bar charts and Network diagram. Project commissioning: mechanical and process.

UNIT III **14 Hours**

Cost Behavior and Profit Planning Marginal Costing; Distinction between Marginal Costing and Absorption Costing; Break-even Analysis, Cost-Volume-Profit Analysis. Various decision-making problems. Standard Costing and Variance Analysis. Pricing strategies: Pareto Analysis. Target costing, Life Cycle Costing. Costing of service sector. Just-in-time approach, Material Requirement Planning, Enterprise Resource Planning, Total Quality Management and Theory of constraints.

UNIT IV **15 Hours**

Activity-Based Cost Management, Bench Marking; Balanced Score Card and Value-Chain Analysis. Budgetary Control; Flexible Budgets; Performance budgets; Zero-based budgets. Measurement of Divisional profitability pricing decisions including transfer pricing.

Quantitative techniques for cost management, Linear Programming, PERT/CPM, Transportation problems, Assignment problems, Simulation, Learning Curve Theory.

Transactional Modes:

Lecture
Case study
Demonstration
Experimentation
Discussion
Problem solving

Suggested Readings:

1. Charles T. Horngren, Srikant M. Datar, Cost Accounting a Managerial Emphasis, Pearson.
2. Ahmed Riahi- Belkaoui., Advanced Management Accounting, Greenwood Publication Group.
3. Robert S Kaplan Anthony A. Alkinson, Management Accounting, Prentice Hall.
4. Ashish K. Bhattacharya, Principles & Practices of Cost Accounting A. H. Wheeler publisher.
5. N.D. Vohra, Quantitative Techniques in Management, Tata McGraw Hill.

Code: CST.600
Course Title: Dissertation/Industrial Project

L	T	P	Cr
0	0	10	5

Course Objectives:

- The student shall have to write his/ her synopsis including an extensive review of literature with simultaneous identification of scientifically sound (and achievable) objectives backed by a comprehensive and detailed methodology. The students shall also present their synopsis to the synopsis approval committee.
- The second objective of Dissertation would be to ensure that the student learns the nuances of the scientific research. Herein the student shall have to carry out the activities/experiments to be completed during Dissertation (as mentioned in the synopsis).

Course Outcome

- The students would present their work to the Evaluation Committee (constituted as per the university rules). The evaluation criteria shall be as detailed below:

Course Contents

Evaluation criteria for Synopsis:

Evaluation Parameter	**Marks**	**Evaluated by**
Review of literature	50	**Internal Evaluation by Dean of School, HOD/ HOD nominee, Two faculty member nominated by Dean/HOD, Supervisor.**
Identification of gaps in knowledge and Problem Statement, Objective formulation & Methodology	50	
Total	**100**	

Student will be given final marks based the average marks by the Evaluation Committee

Timeline Works for Synopsis and Mid-Term:

Month	**JULY**	**AUG**	**SEP**	**OCT**	**NOV**	**DEC**
Synopsis	Bi-Weekly report submitted to Supervisor	Submission of Synopsis and Presentation				
Mid-Term			Bi-Weekly report submitted to Supervisor	Report submission in 3rd week Final Presentation in 4th week	Final Submission of Mid Term Report	

Grading of Marks:

Grades	A	B	C	D	E
Marks	85-100	84-75	74-65	64-40	0-40

Grading Evaluation:

Abbreviations of Grades	**Grades**
Excellent	A
Very Good	B
Good	C
Average	D
Below Average/ Un-Satisfactory	E

Evaluation criteria for Mid-Term:

Evaluation Parameter	**Maximum Marks**	**Evaluated By**
Mid Term Review and Presentation	50	**Internal/External Evaluation by Dean of School, HOD/ HOD nominee, Two faculty member nominated by Dean/ HOD, Supervisor.**
Continuous evaluation	50	
Total	**100**	

Code: CST.559
Course Title: Capstone Lab

L	**T**	**P**	**Cr**
0	0	2	1

In this, the student has to select an area and specify the base paper in that area to implement the same and show the results.
Evaluation criteria will be based on objectives stated and achieved

Course Objective:

- The objective of this lab is to help a team of students develop and execute an innovative project idea under the direction of the Capstone course incharge.

Course Outcome:
After the completion of the course the students will be able to

- Complete the four phases of project development: requirements analysis, design, implementation, and documentation.

Timeline Work:

Month	**AUG**	**SEP**	**NOV**
Seminar	Submit area and Objectives to be achieved	Weekly report to faculty Incharge.	3rd week submit report 4th week Presentation

Evaluation Criteria:

Evaluation Parameter	**Marks**	**Evaluated By**
Area & Objectives	5	**Evaluation Committee**
Reports and Implementation	10	
Presentation and Viva-voce	10	
Total	**25**	

Student will be given final marks based the average marks by the Evaluation Committee

L	T	P	Cr
0	0	16	8

Code: CST.600
Course Title: Dissertation
Course Objectives:

In Dissertation the student shall have to carry out the activities/experiments to be completed during Dissertation (as mentioned in the synopsis).

Course Outcomes:

The students would present their work to the evaluation Committee (constituted as per the university rules).

One research paper (either communicated to a Journal or accepted/ presented/published in conference proceedings) out of the dissertation research work is compulsory. The Evaluation criteria shall be as detailed below:

Evaluation Parameter	Maximum Marks	Evaluated By
Parameters by External Expert (As per University Criteria)	50	**Internal/External Evaluation by Dean of School, DAA Nominee, HOD/ HOD nominee, Supervisor.**
Presentation and defence of research work	50	
Total	100	

Student will be given final marks based the average marks by the Evaluation Committee

Timeline Work of Dissertation:

Month	JAN	FEB	MAR	APR	MAY	JUN
Dissertation	Bi-Weekly report submitted to Supervisor	Bi- Weekly report submitted to Supervisor	Report submission in 1st week	Pre-Submission Presentation in 3st week Report submission in 4th week	Final Submission of Dissertation/ Industrial Project and External Evaluation	

Grading of Marks:

Grades	A	B	C	D	E
Marks	85-100	84-75	74-65	64-40	0-40

Grading Evaluation:

Abbreviations of Grades	Grades
Excellent	A
Very Good	B
Good	C
Average	D
Below Average/ Un-Satisfactory	E