JNTUGV- COLLEGE OF ENGINEERING VIZIANAGARAM

DEPARTMENT OF INFORMATION TECHNOLOGY

B.TECH- IT (R23)

B.Tech. – III Year I Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Advanced Java	3	0	0	3
2	Professional Core	Computer Networks	3	0	0	3
3	Professional Core	Automata Theory & Compiler Design	3	0	0	3
4	Professional Elective-I	1. Object Oriented Analysis and Design 2. Cyber Security 3. Artificial Intelligence 4. Microprocessors & Microcontrollers 5. Data Warehousing & Data Mining 6. 8/12 week MOOC Swayam/ NPTEL course recommended by the BoS	3	0	0	3
5	Open Elective- I	1. Principles of Operating Systems 2. Computer Organization and Architecture	3	0	0	3
6	Professional Core	Advanced Java Lab	0	0	3	1.5
7	Professional Core	Computer Networks Lab	0	0	3	1.5
8	Skill Enhancement course	Full Stack Development 1	0	1	2	2
9	Engineering Science	User Interface Design using Flutter / SWAYAM Plus - Android Application Development (with Flutter)	0	0	2	1
10	Evaluation of Community Service Internship		-	-	-	2
Total			15	1	10	23
MC	Minor Course (Student may select from the same specialized minors pool)		3	0	0	3
MC	Minor Course through SWAYAM/NPTEL (minimum 12 week, 3 credit course)		3	0	0	3
HC	Honors Course (Student may select from the same honors pool)		3	0	0	3
HC	Honors Course (Student may select from the same honors pool)		3	0	0	3

B.Tech. III Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Advanced Data Structures	3	0	0	3
2	Professional Core	Cryptography & Network Security	3	0	0	3
3	Professional Core	Machine Learning	3	0	0	3
4	Professional Elective-II	1. Software Testing Methodologies 2. DevOps 3. Generative AI 4. Intelligent AI Agents 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Professional Elective-III	1. Software Project Management 2. Mobile Adhoc Networks 3. Natural Language Processing 4. Distributed Operating System 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
6	Open Elective – II	Principles of Database Management Systems	3	0	0	3
7	Professional Core	Advanced Data structures Lab	0	0	3	1.5
8	Professional Core	Machine Learning Lab	0	0	3	1.5
9	Skill Enhancement course	Salesforce AI Agent	0	1	2	2
10	Audit Course	Technical Paper Writing & IPR	2	0	0	-
Total			**20**	**1**	**08**	**23**
Mandatory Industry Internship / Mini Project of 08 weeks duration during summer vacation						

MC	Minor Course (Student may select from the same specialized minors pool)	3	0	3	4.5
HC	Honors Course (Student may select from the same honors pool)	3	0	0	3

* Under Industry Internship interested students can pursue SWAYAM Plus courses viz., Hands-on Masterclass on Data Analytics OR Artificial Intelligence for Real-World Application

B.Tech. IV Year I Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Internet of Things	3	0	0	3
2	Management Course- II	Human Resources & Project Management	2	0	0	2
3	Professional Elective-IV	1. Software Architecture & Design Pattern 2. Deep Learning 3. Computer Vision 4. Block chain Technology 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
4	Professional Elective-V	1. Agile methodologies 2. Big Data Analytics 3. Mobile Computing 4. Cyber Physical Systems 5. 8/12 week MOOC Swayam/NPTEL course recommended by the BoS 6. Cloud Computing 7. Intro to LLM	3	0	0	3
5	Open Elective-III	Object-Oriented Programming Through Java	3	0	0	3
6	Open Elective-IV	1.Principles of Software Engineering 2.Computer Networks	3	0	0	3
7	Skill Enhancement Course	Prompt Engineering/ SWAYAM Plus - Certificate program in Prompt Engineering and ChatGPT	0	1	2	2
8	Audit Course	Constitution of India	2	0	0	-
9	Internship	Evaluation of Industry Internship / Mini Project	-	-	-	2

	Total	**19**	**1**	**02**	**21**
MC	Minor Course (Student may select from the same specialized minors pool)	3	0	0	3
HC	Honors Course (Student may select from the same honors pool)	3	0	0	3
HC	Honors Course (Student may select from the same honors pool)	3	0	0	3

B.Tech. IV Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Internship & Project Work	Full semester Internship / Project Work	0	0	24	12

Note : *Students need to do at least ONE MOOC/ NPTEL Course (of 3 credits out of 160 credits) to meet the mandatory requirement (11th criteria, as per R23 Regulations); they are allowed to register one semester in advance*

Open Electives, offered to other department students:

Open Elective I	1. Principles of Operating Systems 2. Computer Organization and Architecture
Open Elective II	Principles of Database Management Systems
Open Elective IIII	Object Oriented Programming Through Java
Open Elective IV	1. Principles of Software Engineering 2. Computer Networks

III Year I Semester	ADVANCED JAVA	L	T	P	C
		3	0	0	3

Course Objectives:

The course aims to:

1. Introduce the architecture and components of J2EE and their role in web development.
2. Familiarize students with servlets and their lifecycle, configuration, and advanced web application handling.
3. Provide a detailed understanding of Java Server Pages (JSP), scripting elements, and web application state management.
4. Enable students to interact with databases using JDBC and build robust, database-driven applications.
5. Introduce the Spring MVC framework and demonstrate how enterprise-level applications can be structured using Spring features such as dependency injection, AOP, and DAO modules.

Course Outcomes:

By the end of the course, students will be able to:

1. **Understand** the architecture of J2EE and the different types of containers and servers used in enterprise web applications.
2. **Develop** web applications using servlets, manage their lifecycle, and implement session tracking mechanisms efficiently.
3. **Design and implement** JSP-based dynamic web pages utilizing scripting, directives, JSTL, and expression language.
4. **Apply** JDBC APIs to interact with relational databases and perform CRUD operations using different statement interfaces and result set operations.
5. **Build** scalable, maintainable enterprise applications using Spring MVC with core features like Bean Factory, Dependency Injection, AOP, and transaction management.

UNIT - I:

J2EE and Web Development: J2EE Architecture Types, J2EE Containers, Types of Servers in J2EE Application, HTTP Protocols and API, Request Processing in Web Application, Web Application Structure, Web Containers and Web Architecture Models.

UNIT - II:

Servlet API and Overview: Servlet Introduction, Servlet Life Cycle(SLC), Types of Servlet, Servlet Configuration with Deployment Descriptor, Working with ServletContext and ServletConfig Object, Attributes in Servelt,, Response and Redirection using Request Dispacher and using sendRedirect Method, Filter API, Manipulating Responses using Filter API, Session Tracking: using Cookies, HTTPSession, Hidden Form Fields and URL Rewriting,Types of Servlet Event: ContextLevel and SessionLevel.

UNIT - III:

Java Server Pages (JSP): Introduction to JSP , Comparison with Servlet, JSP Architecture, JSP: Life Cycle, Scripting Elements, Directives, Action Tags, Implicit Objects, Expression Language(EL), JSP Standard Tag Libraries(JSTL), Custom Tag, Session Management, Exception Handling, CRUD Application

UNIT-IV:

JDBC Programming: JDBC Architecture, Types of JDBC Drivers, Introduction to major JDBC Classes and Interface, Creating simple JDBC Application, Types of Statement (Statement Interface, PreparedStatement, CallableStatement), Exploring ResultSet Operations, Batch Updates in JDBC, Creating CRUD Application, Using Rowsets Objects, Managing Database Transaction.

UNIT-V:

Java Web Frameworks: Spring MVC Spring: Introduction, Architecture, Spring MVC Module, Life Cycle of Bean Factory, Explore: Constructor Injection, Dependency Injection, Inner Beans, Aliases in Bean, Bean Scopes, Spring Annotations, Spring AOP Module, Spring DAO, Database Transaction Management, CRUD Operation using DAO and Spring API.

Text Books:

1. Black Book "Java server programming" J2EE, 1st ed., Dream Tech Publishers, 2008.
2. Complete Reference J2EE, James Keogh, McGraw Hill publication
3. Professional Java Server Programming, Subrahmanyam Allamaraju, Cedric Buest, Wiley Publication
4. Spring in Action, 3rd edition , Craig walls, Manning Publication

Reference Books:

1. Core Java, Volume II: Advanced Features, Cay Horstmann, Gary Cornell Pearson Publication
2. JDBC™ API Tutorial and Reference, Third Edition, Maydene Fisher, Jon Ellis, Jonathan Bruce, Addison Wesley
3. Beginning JSP, JSF and Tomcat, Giulio Zambon, Apress

III Year I Semester	COMPUTER NETWORKS	L	T	P	C
		3	0	0	3

Course Objectives:

The course aims to:

1. Provide a clear understanding of the basic requirements of network hardware, software, and architecture.
2. Familiarize students with the layered architecture of network protocols and the hierarchical structure of physical network infrastructure.
3. Introduce various network interconnecting devices and associated hardware components.
4. Explore advanced networking concepts including wireless and wireless sensor networks.

Course Outcomes (COs):

After successful completion of this course, the student will be able to:

1. Explain the fundamental concepts of computer networks, reference models, and network hardware/software components.
2. Analyze the functionalities of data link and network layers, including error detection, routing algorithms, and IP addressing schemes.
3. Apply appropriate protocols for designing and configuring reliable communication between computers across networks.
4. Evaluate the performance of various transport protocols and network services such as TCP, UDP, DNS, HTTP, and email.
5. Compare and contrast wired and wireless network technologies, devices, and emerging concepts in networking.

UNIT--I:

Introduction to Computer Networks: Introduction, Network Hardware, Network Software, Reference Models,Network Examples, Internet-Based Applications. Architecture of the Internet.

Physical Layer: Guided transmission media, Wireless transmission media, Switching- **Circuit Switching:** Phases:Setup, data transfer, teardown, use in traditional telephone networks, Packet Switching, Store-and-forward, statistical multiplexing, Advantages over circuit switching

UNIT- - II:

Data Link Layer - Design issues, Error Detection & Correction, Elementary Data Link Layer Protocols, Sliding window protocols Multiple Access Protocols - ALOHA, CSMA, CSMA/CD, CSMA/CA, Collision free protocols,

Ethernet- Physical Layer, Ethernet Mac Sub layer,

Data link layer switching: Use of bridges, learning bridges, spanning tree bridges, repeaters, hubs, bridges, switches, routers and gateways.

UNIT- - III:

Network Layer: Network Layer Design issues, store and forward packet switching connection less and connection oriented networks-routing algorithms-optimality principle, shortest path, flooding, Distance Vector Routing, Count to Infinity Problem, Link State Routing, Path Vector Routing, Hierarchical Routing; Congestion control algorithms, IP addresses, CIDR, Subnetting, SuperNetting, IPv4, Packet Fragmentation.

UNIT-- IV:

Transport layer: Transport Service, Elements of Transport Protocols, TCP and UDP Protocols, Quality of Service Model, Best Effort Model, Network Performance Issues.

UNIT- -V:

Application Layer: Overview of DNS, World Wide Web, HTTP Protocols, Electronic Mail, and Firewalls.

Text Books:

1. Computer Networks, Andrew S Tanenbaum. (6th ed.). Pearson Edu.
2. Computer Networking. TOP DOWN APP. Kurose, J. F., & Ross, K. W. (8th ed.). Pearson Education.

References:

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

III Year I Semester	AUTOMATA THEORY & COMPILER DESIGN	L	T	P	C
		3	0	0	3

Course Objectives:

1. To introduce the fundamentals of formal languages, grammars, and finite automata.
2. To design and analyze regular expressions, finite automata (FAs), and pushdown automata (PDAs).
3. To develop an understanding of the theoretical foundations and practical approaches used in compiler construction.
4. To explore lexical analysis, parsing strategies, and syntax-directed translation.
5. To familiarize students with intermediate code generation, code optimization techniques, and target code generation.

Course Outcomes :

By the end of the course, students will be able to:

1. Explain and apply formal language theory, including regular expressions, finite automata, and the Chomsky hierarchy.
2. Design context-free grammars and construct pushdown automata for language recognition.
3. Analyze lexical analysis and implement top-down parsing techniques, including LL(1) parsing and recursive descent.
4. Construct bottom-up parsers using LR, SLR, and LALR methods, and apply syntax-directed translation.
5. Generate intermediate and target code, perform type checking, and apply basic code optimization techniques.

UNIT- – I:

Regular Expressions, Languages and Finite Automata - Formal Languages and the Chomsky Hierarchy, Regular Expressions and Regular Languages, Algebraic Laws for Regular Expressions, Applications of Regular Expressions, Abstract model of Finite Automaton, Transition Tables and Transition Graphs, Deterministic Finite Automata (DFA), Nondeterministic Finite Automata (NFA), Converting NFA to DFA, Finite Automata with ϵ transitions (NFA-ϵ), Converting NFA-ϵ to NFA/DFA, Minimization of Finite Automata, Equivalence of FA and Regular Expressions

UNIT--II:

Context Free Grammars and Push Down Automata: Context Free Grammars (CFG) and Context Free Languages (CFL), Design of CFGs, Leftmost and Rightmost Derivations, Parse Trees, Applications of CFGs, Ambiguity in Grammars and

Languages, Push Down Automata (PDA), The Language of a PDA, Equivalence of PDAs and CFGs

UNIT--III:

Lexical Analysis and Top-Down Parsing-The structure of a compiler, Role of lexical analyzer, Input Buffering, Specification of tokens, Recognition of tokens, The Lexical Analyser Generator –LEX ; Introduction to Syntax Analysis, Eliminating ambiguity and left recursion from a CFG, Recursive Decent Parsing, LL(1) Grammars, Nonrecursive Predictive Parsing

UNIT--IV:

Bottom-Up Parsing and Syntax Directed Translation- Shift-Reduce Parsing, Simple LR parsing, Canonical LR(1) Parsing, LALR Parsing, Parser Generators , Syntax Directed Definitions, Evaluation Orders for SDDs, Syntax Directed Translation Schemes

UNIT--V:

Intermediate Code Generation, Code Generation and Optimization: Three address code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Issues in the design of a Code Generator, The Target Language, A simple Code Generator Basic Blocks and Flow Graphs, Optimization of Basic Blocks, Peephole Optimization

Textbooks:

1. Introduction to Automata Theory, Languages and Computation, J.E.Hopcroft, R.Motwani and J.D.Ullman, 3rd Edition, Pearson, 2008.
2. Compilers Principles, Techniques and Tools, 2nd Edition, Alfred V.Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson

Reference Books:

1. Introduction to Languages and The Theory of Computation, John C. Martin, McGraw Hill.
2. Theory of Computer Science-Automata, Languages and Computation, K.L.P.Mishra and N.Chandrasekaran, 3rd Edition, PHI, 2007
3. Compiler Construction‖, K.V.N. SUNIT-ha, Pearson, 2013
4. Compiler Design, SandeepSaxena, Rajkumar Singh Rathore, S.Chand publication

III Year I Semester	OBJECT ORIENTED ANALYSIS AND DESIGN (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

1. Become familiar with all phases of Object-Oriented Analysis and Design (OOAD).
2. Master the key features and diagrams of the Unified Modeling Language (UML).
3. Understand and apply object-oriented concepts and technologies to solve real-world problems across various domains.
4. Learn and apply object design principles to guide effective implementation of software systems.

Course Outcomes:

By the end of the course, students will be able to:

1. Understand the structure and complexity of software systems and design solutions using object-oriented principles.
2. Create and interpret UML models, including class, object, and package diagrams, to represent static system structure.
3. Model system behavior using use cases, interaction diagrams, and activity diagrams based on functional requirements.
4. Develop advanced behavioral and architectural models, including state machines, component, and deployment diagrams.
5. Apply OOAD techniques and UML modeling in practical case studies and real-world application domains.

UNIT-- I:

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems. **Case Study:** System Architecture: Satellite-Based Navigation

UNIT- -II:

Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. **Basic Structural Modeling:** Classes, Relationships, common Mechanisms, and diagrams. **Case Study:** Control System: Traffic Management.

UNIT-- III:

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams. **Advanced Structural Modeling:** Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **Case Study:** AI: Cryptanalysis.

UNIT- -IV:

Basic Behavioral Modeling-I: Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. **Case Study:** Web Application: Vacation Tracking System

UNIT-- V:

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams. **Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams. **Case Study:** Weather Forecasting

Text Books:

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston , “Object- Oriented Analysis and Design with Applications”, 3rd edition, 2013, PEARSON.
2. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.

Reference Books:

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

III Year I Semester	CYBER SECURITY (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

The course aims to:

1. Introduce the fundamental concepts, objectives, and roles in cyber security.
2. Provide understanding of security architecture and lifecycle management in information security.
3. Equip students with knowledge of incident response, operational security measures, and risk management.
4. Familiarize students with threat detection, monitoring tools, and traffic analysis techniques.
5. Introduce backdoor systems, metasploit framework, and methods to secure operating systems.

Course Outcomes:

1. Understand the fundamental principles, objectives, and roles of cyber security, and differentiate it from information security.
2. Apply concepts of security architecture, risk management, and lifecycle management to identify and mitigate risks and vulnerabilities in information systems.
3. Analyze and implement incident response strategies, operational security practices, and configuration management to secure digital assets and network infrastructure.
4. Utilize monitoring tools and techniques for threat detection, log analysis, and network traffic evaluation to maintain security posture.
5. Demonstrate knowledge of backdoor systems, penetration testing tools like Metasploit, and system hardening techniques to enhance system security.

UNIT--I:

Introduction to Cyber security- Cyber security objectives, Cyber security roles, Differences between Information Security & Cyber security, Cyber security Principles Confidentiality, integrity, &availability Authentication & non- repudiation.

UNIT--II:

Information Security (IS) within Lifecycle Management-Lifecycle management landscape, Security architecture processes, Security architecture tools, Intermediate lifecycle management concepts, Risks & Vulnerabilities-Basics of risk management, Operational threat environments, Classes of attacks.

UNIT--III:

Incident Response- Incident categories, Incident response Incident recovery, and Operational security protection: Digital and data assets, ports and protocols, Protection technologies, Identity and access Management, configuration management.

UNIT--IV:

Threat Detection and Evaluation (DE): Monitoring- Vulnerability Management, Security Logs and Alerts, Monitoring Tools and Appliances. Analysis- Network traffic Analysis, packet capture and analysis

UNIT--V:

Introduction to backdoor System and security-Introduction to metasploit, Backdoor, demilitarized zone(DMZ),Digital Signature, Brief study on Harding of operating system.

Text Books:

1. NASSCOM: Security Analyst Student Hand Book.

2. Information Security Management Principles Updated Edition by David Alexander, Amanda Finch, David Sutton ,Published by BCS.

Reference Books:

1. CSX- cyber security fundamentals , Published by ISACA, Cyber security, Network Security, Data Governance Security.

E-Resources:

1. CERT-In Guidelines- http://www.cert-in.org.in/
2. https://www.coursera.org/learn/introduction-cybersecurity-cyber-attacks [Online Course]
3. https://computersecurity.stanford.edu/free-online-videos[Free Online Videos]
4. Nickolai Zeldovich. 6.858 Computer Systems Security. Fall 2014. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu License: Creative Commons BY-NC-SA.

III Year I Semester	ARTIFICIAL INTELLIGENCE (Professional Elective-I)	L	T	P	C
		3	0	0	3

Pre-requisite:

1. Knowledge in Computer Programming.
2. A course on "Mathematical Foundations of Computer Science".
3. Background in linear algebra, data structures and algorithms, and probability.

Course Objectives:

1. To introduce students to the fundamental concepts and goals of Artificial Intelligence.
2. To teach various AI techniques for solving complex and real-world problems.
3. To familiarize students with the structure and functioning of Expert Systems.
4. To explore AI applications such as game playing, machine learning, and automated theorem proving.
5. To understand and apply different knowledge representation and reasoning techniques.

Course Outcomes :

By the end of the course, students will be able to:

1. Explain the foundational principles and history of Artificial Intelligence, and describe intelligent agents and their environments.
2. Apply various search strategies (uninformed and informed) and adversarial search techniques like minimax and alpha-beta pruning.
3. Represent and reason with knowledge using predicate logic, semantic networks, rules, and probabilistic models.
4. Use logical inference techniques such as forward/backward chaining and resolution, and apply learning methods like decision trees and reinforcement learning.
5. Understand the architecture and functioning of expert systems, including typical examples like MYCIN and DART.

UNIT - I

Introduction: AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

UNIT- II

Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A*

,AO* Algorithms, Problem reduction, Game Playing-Adversial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

UNIT - III

Representation of Knowledge: Knowledge representation issues, predicate logic- logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules based deduction systems. Reasoning under uncertainty, review of probability, Bayes' probabilistic interferences and dempstershafer theory.

UNIT - IV

Logic concepts: First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning.

UNIT - V

Expert Systems: Architecture of expert systems, Roles of expert systems – Knowledge Acquisition Meta knowledge Heuristics. Typical expert systems – MYCIN, DART, XCON: Expert systems shells.

Textbooks:

1. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", SecondEdition, Pearson Education.
2. Kevin Night and Elaine Rich, Nair B., "Artificial Intelligence (SIE)", Mc Graw Hill

Reference Books:

1. David Poole, Alan Mackworth, Randy Goebel,"Computational Intelligence: a logical approach", Oxford University Press.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problemsolving", Fourth Edition, Pearson Education.
3. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers.
4. Artificial Intelligence, SarojKaushik, CENGAGE Learning.

Online Learning Resources:

1. https://ai.google/
2. https://swayam.gov.in/nd1_noc19_me71/preview

III Year I Semester	MICROPROCESSORS & MICROCONTROLLERS (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

1. To introduce the fundamental architectural concepts of microprocessors and microcontrollers.
2. To impart knowledge of the addressing modes and instruction sets of 8086 and 8051.
3. To develop an understanding of assembly language programming concepts.
4. To explain memory and I/O interfacing techniques using 8086 and 8051.
5. To provide an overview of 16-bit and 32-bit microcontrollers, including comparisons with modern architectures like PIC and ARM.

Course Outcomes (COs):

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

1. Explain the internal architecture, operation modes, and interrupt handling of the 8086 microprocessor.
2. Develop assembly language programs for 8086 using various addressing modes and assembler directives.
3. Interface memory and peripheral devices with 8086 using chips like 8255, 8251, 8237A, and 8259, and write relevant I/O programs.
4. Describe the architecture, instruction set, and assembly programming techniques of the 8051 microcontroller.
5. Design and implement real-time interfacing applications using 8051 for devices like LCDs, keypads, sensors, ADC/DACs, and motors, and compare 8051 with PIC and ARM microcontrollers.

UNIT- I:

8086 Architecture: Main features, pin diagram/description, 8086 microprocessor family, internal architecture, bus interfacing UNIT-, execution UNIT-, interrupts and interrupt response, 8086 system timing, minimum mode and maximum mode configuration.

UNIT- II:

8086 Programming: Program development steps, instructions, addressing modes, assembler directives, writing simple programs with an assembler, assembly language program development tools.

UNIT- III:

8086 Interfacing: Semiconductor memories interfacing (RAM, ROM), Intel 8255 programmable peripheral interface, Interfacing switches and LEDS, Interfacing seven segment displays, software and hardware interrupt applications, Intel 8251 USART architecture and interfacing, Intel 8237a DMA controller, stepper motor, A/D and D/A converters, Need for 8259 programmable interrupt controllers.

UNIT- IV:

Microcontroller, Architecture of 8051, Special Function Registers(SFRs), I/O Pins Ports and Circuits, Instruction set, Addressing modes, Assembly language programming.

UNIT- V:

Interfacing Microcontroller, Programming 8051 Timers, Serial Port Programming, Interrupts Programming, LCD & Keyboard Interfacing, ADC, DAC & Sensor Interfacing, External Memory Interface, Stepper Motor and Waveform generation, Comparison of Microprocessor, Microcontroller, PIC and ARM processors

Textbooks:

1. Microprocessors and Interfacing – Programming and Hardware by Douglas V Hall, SSSP Rao, Tata McGraw Hill Education Private Limited, 3rd Edition,1994.
2. K M Bhurchandi, A K Ray, Advanced Microprocessors and Peripherals, 3rd edition, McGraw Hill Education, 2017.
3. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, 2nd edition, Pearson, 2012.

Reference Books:

1. Ramesh S Gaonkar, Microprocessor Architecture Programming and Applications with the 8085, 6th edition, Penram International Publishing, 2013.
2. Kenneth J. Ayala, The 8051 Microcontroller, 3rd edition, Cengage Learning, 2004.

III Year I Semester	DATA WAREHOUSING & DATA MINING (Professional Elective-I)	L	T	P	C
		3	0	0	3

Pre-requisites: Data Structures, Algorithms, Probability & Statistics, Data Base Management Systems

Course Objectives:

The main objectives of this course are to:

1. Introduce the basic concepts and techniques of data warehousing and data mining.
2. Examine the types of data to be mined and apply suitable preprocessing methods on raw data.
3. Enable students to discover interesting patterns using classification, clustering, and association rule mining.
4. Analyze supervised and unsupervised learning models and evaluate the performance of data mining algorithms.
5. Explore data mining applications and technologies with a focus on cloud-based data warehousing and visualization.

Course Outcomes:

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

1. Explain the fundamental concepts of data warehousing, OLAP, and data mining, including data types and similarity measures.
2. Apply data preprocessing techniques such as cleaning, integration, reduction, and transformation to prepare data for mining tasks.
3. Implement and evaluate classification models using decision trees, Bayesian classifiers, and rule-based methods.
4. Generate and analyze association rules using Apriori and FP-Growth algorithms to uncover interesting relationships in large datasets.
5. Perform clustering analysis using algorithms such as K-Means, Hierarchical Clustering, and DBSCAN, and evaluate their strengths and weaknesses.

UNIT--I:

Data Warehousing and Online Analytical Processing: Basic concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Cloud Data Warehouse, Data Mining and Patten Mining, Technologies, Applications, Major issues, Data Objects & Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity. (Text Book- 1)

UNIT- II:

Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization. (Text Book- 1)

UNIT--III:

Classification: Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction, Visual Mining for Decision Tree Induction, Bayesian Classification Methods: Bayes Theorem, Naïve Bayes Classification, Rule-Based Classification, Model Evaluation and Selection. (Text Book- 2)

UNIT--IV:

Association Analysis: Problem Definition, Frequent Itemset Generation, Rule Generation: Confident Based Pruning, Rule Generation in Apriori Algorithm, Compact Representation of frequent item sets, FP-Growth Algorithm. (Text Book- 2)

UNIT--V:

Cluster Analysis: Overview, Basics and Importance of Cluster Analysis, Clustering techniques, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bi-secting K Means, Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. (Text Book- 2)

Text Books:

1. Data Mining concepts and Techniques, 3rd edition, Jiawei Han, Michel Kamber, Elsevier, 2011.
2. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson, 2012.

Reference Books:

1. Data Mining: VikramPudi and P. Radha Krishna, Oxford Publisher.
2. Data Mining Techniques, Arun K Pujari, 3rd edition, Universities Press,2013.
3. (NPTEL course by Prof.PabitraMitra) http://onlinecourses.nptel.ac.in/noc17_mg24/preview
4. http://www.saedsayad.com/data_mining_map.htm

III Year I Semester	ADVANCED JAVA LAB	L	T	P	C
		0	0	3	1.5

Course Objectives: the main objectives of the course are

- To make use of Servlet and JSP API in the process of enterprise application deployment.
- Implement components such as JSTL
- Distinguish Application Server, Web Container, JDBC
- Design and Development of web application having collaboration of Servlets, JSPs, Spring

Lab should cover the following concepts:

- JDBC programming
- J2EE and Web development
- Servlets
- Java Server Pages
- Java Web Frameworks

Sample List of Experiments:

1. Write a JDBC application which will interact with Database and perform the following task.
    a. Create Student Table with RollNo, Name, and Address field and insert few records.
    b. Using Statement Object display the content of Record.
    c. Using Statement Object Insert Two Record.
    d. Using Statement Object Update One Record.
    e. Using Statement Object Delete One Record.
    f. Using Statement Object display the content of Record.

2. Write a JDBC application which will interact with Database and perform the following task.
    a. Create Student Table with RollNo, Name, and Address field and insert few records.
    b. Using PreparedStatement Object display the content of Record.
    c. Using PreparedStatement Object Insert Two Record.
    d. Using PreparedStatement Object Update One Record.
    e. Using PreparedStatement Object Delete One Record.
    f. Using PreparedStatement Object display the content of Record

3. Write a JDBC application which will interact with Database and perform the following task.

   a. Create a store procedure which will insert one record into employee table.
   b. Create a store procedure which will retrieve salary for given employee id.
   c. Write a java application which will call the above procedure and display appropriate information on screen

4. Design a JDBC application which will demonstrate Scrollable ResultSet functionality.

5. Design a JDBC application which will demonstrate Updatable ResultSet functionality.

6. Write down the Program for testing the Servlet and study deployment descriptor.

7. Write down the program for testing the include action for servlet collaboration.
8. Create login form and perform state management using Cookies, HttpSession and URL Rewriting.

9. Write down the Program which displays the simple JSP file

10. Write down the program in which input the two numbers in an html file and then display the addition in JSP file.

11. Perform Database Access through JSP.

12. Write down a program which demonstrates the core tag of JSTL.

13. Write down a program which demonstrates the Format tag of JSTL.

14. Write down a program which demonstrates the Function tag of JSTL.

15. Write down a program which demonstrates the SQL tag of JSTL.

16. Study and Implement MVC using Spring Framework

17. Using Spring Template manage Database and Transaction.

III Year I Semester	Computer Networks Lab	L	T	P	C
		0	0	3	1.5

Course Objectives:

The objectives of this lab course are to:

1. Provide hands-on experience in configuring and analyzing computer networks.
2. Familiarize students with framing methods, error detection/correction techniques, and flow control mechanisms.
3. Enable students to simulate and implement various routing and congestion control algorithms.
4. Introduce network traffic monitoring using tools such as Wireshark.
5. Explore network behavior through simulation tools like NS2/NS3.

Course Outcomes (COs):

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

1. Identify and configure basic network components and apply framing techniques such as character and bit stuffing.
2. Implement and analyze error detection and correction techniques including Hamming Code and various CRC variants.
3. Simulate and evaluate medium access and flow control protocols such as Stop-and-Wait, Go-Back-N, and Selective Repeat.
4. Demonstrate congestion control and routing algorithms such as Leaky Bucket, Dijkstra's, and Distance Vector Routing using simulation or programming.
5. Analyze network traffic using Wireshark and evaluate packet behavior and transmission through NS2/NS3 simulations.

Category 1: Network Basics and Layered Protocols (Foundational)

- Study of Network Devices and LAN Setup
- Data Link Layer Framing Methods – Character Stuffing and Bit Stuffing
- Checksum Generation and Verification

Category 2: Error Detection and Correction

- Hamming Code Implementation
- Cyclic Redundancy Check (CRC) – CRC-12, CRC-16, CRC-CCITT

Category 3: Medium Access & Flow Control

- Stop-and-Wait Protocol Implementation
- Sliding Window Protocol – Go-Back-N
- Sliding Window Protocol – Selective Repeat

Category 4: Congestion and Routing Algorithms

- Leaky Bucket Algorithm for Congestion Control
- Dijkstra's Algorithm for Shortest Path Routing

- Distance Vector Routing Protocol Simulation

Category 5: Traffic Analysis and Tools

- Introduction to Wireshark – Capturing and Analyzing Packets on a Local Network
- Wireshark Packet Analysis – Filtering and Statistics

Category 6: Network Simulation with NS2/NS3

- Simulation of TCP/UDP Behavior and Packet Drop in NS2

Online Resources

- Wireshark Official Labs: https://wiki.wireshark.org/
  Includes tutorials and filters for TCP, HTTP, and DNS.
- NS2/NS3 Documentation and Tutorials:
  - o The Network Simulator - ns-2
  - o ns-3 Tutorial — Tutorial
- NPTEL Course on Computer Networks (by IIT Professors):
  Computer Networks and Internet Protocol - Course

III Year I Semester	Full Stack Development (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

Course Objectives:

By the end of the course, students will be able to:

- Understand the architecture and components of web development.
- Develop well-structured web pages using HTML and HTML5.
- Create interactive forms and frame-based layouts.
- Integrate multimedia and ensure web accessibility.
- Style web pages effectively using CSS.
- Incorporate JavaScript for interactivity.
- Design and build dynamic user interfaces using React.js.
- Build single-page applications with React Router and manage state using Redux or Context API.
- Integrate frontend applications with RESTful APIs.
- Create backend servers and RESTful APIs using Node.js, Express.js and design MongoDB databases.
- Develop and integrate full-stack MERN applications

UNIT--I :

Web Essentials: Clients, Servers, and Communication. The Internet-Basic Internet Protocols The World Wide Web-HTTP request message-response message-Web Clients Web Servers. Markup Languages: XHTML an Introduction to HTML, History, Versions, Basic, XHTML Syntax and Semantics Some Fundamental HTML Elements-Relative URLs-Lists-tablesFrames-Forms.

UNIT--II:

Style Sheets: CSS-Introduction to Cascading Style Sheets-Features-Core Syntax-Style Sheets and HTML- Style Rule Cascading and Inheritance-Text Properties-Box Model Normal Flow Box Layout beyond the Normal Flow-CSS3.0, Basic Introduction to Java Script.

UNIT--III:

Java Script, React JS Introduction to React, Obstacles and Roadblocks, keeping Up with the Changes, Working with the Files, Pure React, Page Setup, The Virtual DOM, React Elements, ReactDOM, Children, Constructing Elements with Data, React Components, DOM Rendering, Factories.

UNIT- -IV :

Node JS Fundamentals: Event Driven Architecture, Modules, HTTP Module, Express JS: Setting Severs, Routing, Middle Ware, RESTful APIs (GET.POST,PUT,DELETE).

UNIT--V :

MongoDB and MERN Integration, MongoDB Basics: NOSQL Concepts, Collections, Documents, CRUD Operations, Mongoose: Schemas, Models, Querying Data. MERN Integration: Connecting MongoDB, Express JS, React JS, Node JS.

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson, 2013.
2. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11)
3. JavaScript: The Good Parts, Douglas Crockford,1st Edition (2008), O'Reilly Media.
4. Web Development with Node and Express, Ethan Brown, 2nd Edition (2019), O'Reilly.

Reference Text Books:

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.
2. Learning React: Modern Patterns for Developing React Apps, Alex Banks, Eve Porcello, 2nd Edition (2020), O'Reilly.
3. MongoDB: The Definitive Guide, Shannon Bradshaw, Kristina Chodorow, 3rd Edition (2019), O'Reilly.

Web Links:

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css
3. https://www.w3schools.com/js/
4. https://www.w3schools.com/jquery/default.asp
5. https://www.w3schools.com/react/default.asp
6. https://www.w3schools.com/nodejs/default.asp
7. https://apidog.com/blog/how-to-create-a-rest-api-with-node-js-and-express/?utm_source=google_dsa&utm_medium=s&utm_campaign=22062217351&utm_content=169453484141&utm_term=&gad_source=5&gad_campaignid=22062217351&gclid=EAIaIQobChMI0O760__1jQMVd6RmAh0xRwTxEAAYASAAEgLscPD_BwE

III Year I Semester	USER INTERFACE DESIGN USING FLUTTER	L	T	P	C
		0	0	2	1

Course Objectives:

- Learns to Implement Flutter Widgets and Layouts
- Understands Responsive UI Design and with Navigation in Flutter
- Knowledge on Widges and customize widgets for specific UI elements, Themes
- Understand to include animation apart from fetching data

List of Experiments:

Students need to implement the following experiments

1 a) Install Flutter and Dart SDK.
b) Write a simple Dart program to understand the language basics.

2 a) Explore various Flutter widgets (Text, Image, Container, etc.).
b) Implement different layout structures using Row, Column, and Stack widgets.

3 a) Design a responsive UI that adapts to different screen sizes.
b) Implement media queries and breakpoints for responsiveness.

4 a) Set up navigation between different screens using Navigator.
b) Implement navigation with named routes.

5 a) Learn about stateful and stateless widgets.
b) Implement state management using set State and Provider.

6 a) Create custom widgets for specific UI elements.
b) Apply styling using themes and custom styles.

7 a) Design a form with various input fields.
b) Implement form validation and error handling.

8 a) Add animations to UI elements using Flutter's animation framework.
b) Experiment with different types of animations (fade, slide, etc.).

9 a) Fetch data from a REST API.
b) Display the fetched data in a meaningful way in the UI.

10 a) Write UNIT- tests for UI components.
b) Use Flutter's debugging tools to identify and fix issues.

Text Books:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.
2. Rap Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps 1st Edition, Apres
3. Richard Rose, Flutter & Dart Cookbook, Developing Full stack Applications for the Cloud, Oreilly.

III Year II Semester	ADVANCED DATA STRUCTURES	L	T	P	C
		3	0	0	3

Course Objectives

1. Understand fundamental and advanced data structures, their operations, and complexity analysis.
2. Analyze algorithms using time and space complexity, recurrence relations, and amortized techniques.
3. Apply advanced data structures like balanced trees, heaps, and disjoint sets to real-world problems.
4. Explore efficient indexing and searching using hashing techniques and trie-based structures.
5. Utilize graph data structures and algorithms for solving computational problems in various domains.

Course Outcomes

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

1. Demonstrate proficiency in using linear and non-linear data structures for problem-solving.
2. Analyze the performance of algorithms using complexity measures, recurrence relations, and amortized analysis.
3. Implement and apply balanced tree structures and advanced heap data structures in algorithm design.
4. Design efficient indexing and search solutions using tries, hashing, and suffix structures.
5. Solve real-world problems using appropriate graph representations and algorithms including MST and shortest path algorithms.

UNIT- I :

Introduction and Algorithm Analysis-Review of basic data structures – arrays, stacks, queues, linked lists, trees, and their operations. Time and space complexity analysis. Recurrence relations and solving them using the Master Theorem. Amortized analysis – aggregate, accounting, and potential methods.

UNIT- II :

Balanced Trees and Indexing Structures-Advanced tree structures – AVL Trees, Red-Black Trees, Splay Trees. B-Trees and B+ Trees for indexing in databases and file systems. Segment Trees and Fenwick Trees (Binary Indexed Trees) for efficient range query processing.

UNIT- III:

Priority Queues and Disjoint Sets- Binary Heaps, Binomial Heaps, Fibonacci Heaps – operations and efficiency. Applications in priority queues and Dijkstra's algorithm. Disjoint Set ADT – union by rank and path compression. Applications in Kruskal's algorithm and network connectivity problems.

UNIT - IV:

Hashing and Tries-Advanced hashing techniques – open addressing, double hashing, cuckoo hashing. Hash functions and collision resolution. Tries, compressed tries, suffix tries, and ternary search trees. Applications in text search and dictionary implementations. Introduction to suffix arrays and Bloom filters.

UNIT- V:

Graph Data Structures and Applications-Representation of graphs – adjacency matrix, list, and edge list. Traversal algorithms – BFS, DFS, and their applications. Advanced concepts – shortest path (Dijkstra, Bellman-Ford), MST (Prim, Kruskal), and Union-Find. Real-world applications in networks, compilers, and social media analysis.

Textbooks

1. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C++*, Pearson, 4th Edition
2. Thomas H. Cormen et al., *Introduction to Algorithms*, MIT Press, 3rd Edition

Reference Books

- Alfred Aho, Jeffrey Ullman, *Data Structures and Algorithms*
- S. Sahni and E. Horowitz, *Fundamentals of Data Structures*
- Steven S. Skiena, *The Algorithm Design Manual*

III Year II Semester	CRYPTOGRAPHY & NETWORK SECURITY	L	T	P	C
		3	0	0	3

Course Objectives

The primary objectives of this course are to:

1. Understand the **fundamental principles** and **mathematical foundations** of cryptography.
2. Explore **symmetric** and **asymmetric encryption** techniques and their cryptographic algorithms.
3. Study the design and applications of **cryptographic hash functions**, **digital signatures**, and **key management protocols**.
4. Learn about **authentication protocols** and **secure communication standards** like Kerberos, IPsec, SSL, and TLS.
5. Examine real-world applications of cryptography in **network security**, **system security**, and **secure messaging**.

Course Outcomes

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

1. Explain the goals of cryptography and classify various attacks, services, and mechanisms.
2. Analyze and apply symmetric encryption techniques such as DES and AES along with the supporting algebraic structures.
3. Demonstrate the working of asymmetric encryption algorithms like RSA, ElGamal, and ECC, and understand the number-theoretic concepts underlying them.
4. Assess the integrity and authentication mechanisms through hash functions, digital signatures, and key management protocols.
5. Apply cryptographic concepts in network security protocols at different layers , and explain modern threats and system-level security techniques like firewalls andIDS.

UNIT- I:

Basic Principles: Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography- integer arithmetic, modular arithmetic, matrices, linear conguence.

UNIT- II:

Symmetric Encryption: Mathematics of Symmetric Key Cryptography-algebraic structures, GF(2^n) Fields, Introduction to Modern Symmetric Key Ciphers-modern block ciphers, modern stream ciphers, Data Encryption Standard- DES structure, DES analysis, Security of DES, Multiple DES, Advanced Encryption Standard-transformations, key expansions, AES ciphers, Analysis of AES.

UNIT- III:

Asymmetric Encryption: Mathematics of Asymmetric Key Cryptography-primes, primality testing, factorization, CRT, Asymmetric Key Cryptography- RSA crypto system, Rabin cryptosystem, Elgamal Crypto system, ECC

UNIT- IV:

Data Integrity, Digital Signature Schemes & Key Management : Message Integrity and Message Authentication-message integrity, Random Oracle model, Message authentication, Cryptographic Hash Functions-whirlpool, SHA-512, Digital Signature-process, services, attacks, schemes, applications, Key Management-symmetric key distribution, Kerberos.

UNIT- V:

Network Security-I: Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS, **Network Security-II :** Security at the Network Layer: IPSec-two modes, two security protocols, security association, IKE, ISAKMP, System Security-users, trust, trusted systems, buffer overflow, malicious software, worms, viruses, IDS, Firewalls.

Text Books:

1. Cryptography and Network Security, 3rd Edition Behrouz A Forouzan, Deb deep Mukhopadhyay, McGraw Hill,2015
2. Cryptography and Network Security,4th Edition, William Stallings, (6e) Pearson,2006
3. Everyday Cryptography, 1st Edition, Keith M.Martin, Oxford,2016

Reference Books:

1. Network Security and Cryptography, 1st Edition, Bernard Meneges, Cengage Learning,2018

III Year II Semester	MACHINE LEARNING	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of the course are to:

1. Define the foundational concepts of **machine learning**, including supervised, unsupervised, and reinforcement learning paradigms.
2. Understand and apply **distance-based algorithms** like **K-Nearest Neighbors (KNN)** for classification and regression.
3. Explore decision tree-based models including **Random Forests** and **Naïve Bayes Classifier**, and analyze their behavior.
4. Gain insight into **linear models** such as **Perceptron**, **Logistic Regression**, and **Support Vector Machines**, along with neural network basics.
5. Implement various **clustering techniques**, including **K-means**, **Fuzzy C-means**, **Rough Clustering**, and **EM-based Clustering**.

Course Outcomes:

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

1. Explain the key concepts, types, and stages of machine learning and their real-world applications.
2. Apply and evaluate proximity-based algorithms like **KNN** for classification and regression problems.
3. Build and analyze decision trees, **Random Forests**, and **Naïve Bayes classifiers** for classification tasks.
4. Implement linear classifiers like **Perceptron**, **SVM**, and **Logistic Regression** and understand neural network training using backpropagation.
5. Demonstrate the ability to apply unsupervised learning algorithms for clustering tasks using **K-Means**, **Fuzzy C-Means**, **Spectral Clustering**, and others.

UNIT--I:

Introduction to Machine Learning: Evolution of Machine Learning, Paradigms for ML, Learning by Rote, Learning by Induction, Reinforcement Learning, Types of Data, Matching, Stages in Machine Learning, Data Acquisition, Feature Engineering, Data Representation, Model Selection, Model Learning, Model Evaluation, Model Prediction, Search and Learning, Data Sets.

UNIT--II:

Nearest Neighbor-Based Models: Introduction to Proximity Measures, Distance Measures, Non-Metric Similarity Functions, Proximity Between Binary Patterns, Different Classification Algorithms Based on the Distance Measures ,K-Nearest Neighbor Classifier, Radius Distance Nearest Neighbor Algorithm, KNN Regression, Performance of Classifiers, Performance of Regression Algorithms.

UNIT--III:

Models Based on Decision Trees: Decision Trees for Classification, Impurity Measures, Properties, Regression Based on Decision Trees, Bias–Variance Trade-off, Random Forests for Classification and Regression. The Bayes Classifier: Introduction to the Bayes Classifier, Bayes' Rule and Inference, The Bayes Classifier and its Optimality, Multi-Class Classification, Class Conditional Independence and Naive Bayes Classifier (NBC)

UNIT--IV:

Linear Discriminants for Machine Learning: Introduction to Linear Discriminants, Linear Discriminants for Classification, Perceptron Classifier, Perceptron Learning Algorithm, Support Vector Machines, Linearly Non-Separable Case, Non-linear SVM, Kernel Trick, Logistic Regression, Linear Regression, Multi-Layer Perceptrons (MLPs), Backpropagation for Training an MLP.

UNIT--V:

Clustering : Introduction to Clustering, Partitioning of Data, Matrix Factorization, Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, Rough Clustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering.

Text Books:

1. "Machine Learning Theory and Practice", M N Murthy, V S Ananthanarayana, Universities Press (India), 2024

Reference Books:

1. "Machine Learning", Tom M. Mitchell, McGraw-Hill Publication, 2017
2. "Machine Learning in Action",Peter Harrington, DreamTech
3. "Introduction to Data Mining", Pang-Ning Tan, Michel Stenbach, Vipin Kumar, 7th Edition, 2019.

III Year II Semester	SOFTWARE TESTING METHODOLOGIES (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives

The main objectives of this course are to:

1. Understand the **fundamental concepts** of software testing and its role in software quality assurance.
2. Introduce different **testing models**, types of bugs, and a taxonomy to classify bugs.
3. Learn and apply **flow graph-based testing**, including **path testing** and **data flow testing**.
4. Explore **domain testing**, **logic-based testing**, and **state transition testing** techniques.
5. Study advanced testing topics like **graph matrices**, **regular expressions**, and their application in tool-based testing.
6. Provide hands-on exposure to **automated software testing tools** such as **JMeter**, **Selenium**, **SoapUI**, or **Catalon**.

Course Outcomes:

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

1. Apply **path testing techniques** using flow graphs and predicates to design test cases.
2. Perform **transaction flow testing**, **data flow testing**, and **domain testing** to assess program correctness and detect anomalies.
3. Analyze software behavior using **path expressions**, **logic-based testing**, and **decision tables**.
4. Apply **state-based and transition testing** for modeling and verifying software systems.
5. Use **graph matrices**, **node reduction algorithms**, and **automated tools** for effective test design and execution.

UNIT - I

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT - II

Transaction Flow Testing: transaction flows, transaction flow testing techniques.

Data Flow testing: Basics of data flow testing, strategies in data flow testing, application of data flow testing.
Domain Testing: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT - III

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.
Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

UNIT - IV

State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.

UNIT - V

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter/selenium/soapUI/Catalon).

Text Books:

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.

Reference Books:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

III Year II Semester	AUGMENTED REALITY & VIRTUAL REALITY (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of this course are to:

1. Provide a strong **foundation in the concepts, history, and components** of Augmented Reality (AR).
2. Introduce students to the **fundamentals of Virtual Reality (VR)** including its technical, perceptual, and physiological aspects.
3. Explain the **hardware and software components** used in AR/VR, such as displays, tracking systems, and computer vision techniques.
4. Describe the **human sensory and perceptual systems**, and how they influence AR/VR system design.
5. Expose students to **interaction techniques** and **audio-visual rendering** used in immersive systems.

Course Outcomes:

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

1. Explain the fundamental concepts, history, components, and tracking technologies involved in Augmented Reality systems.
2. Apply computer vision techniques for AR, understand interaction modalities, and describe AR software architectures.
3. Understand the historical context and mathematical modeling of Virtual Reality including viewing geometry, optics, and light behavior.
4. Analyze human visual physiology and visual perception for realistic rendering and immersive experiences in VR.
5. Apply concepts of motion, interaction, and auditory rendering to enhance immersion in virtual environments.

UNIT– I:

Introduction to Augmented Reality: Augmented Reality - Defining augmented reality, history of augmented reality, Examples, Related fields

Displays: Multimodal Displays, Visual Perception, Requirements and Characteristics, Spatial Display Model, Visual Displays

Tracking: Tracking, Calibration, and Registration, Coordinate Systems, Characteristics of Tracking Technology, Stationary Tracking Systems, Mobile Sensors

UNIT – II:

Computer Vision for Augmented Reality: Marker Tracking, Multiple-Camera Infrared Tracking, Natural Feature Tracking by Detection, Outdoor Tracking.

Interaction: Output Modalities, Input Modalities, Tangible Interfaces, Virtual User Interfaces on Real Surfaces, Augmented Paper, Multi-view Interfaces, Haptic Interaction

Software Architectures: AR Application Requirements, Software Engineering Requirements, Distributed Object Systems, Dataflow, Scene Graphs

UNIT – III:

Introduction to Virtual Reality: Defining Virtual Reality, History of VR, Human Physiology and Perception

The Geometry of Virtual Worlds: Geometric Models, Axis-Angle Representations of Rotation, Viewing Transformations

Light and Optics: Basic Behavior of Light, Lenses, Optical Aberrations, The Human Eye, Cameras, Displays

UNIT – IV:

The Physiology of Human Vision: From the Cornea to Photoreceptors, From Photoreceptors to the Visual Cortex, Eye Movements, Implications for VR

Visual Perception: Visual Perception - Perception of Depth, Perception of Motion, **Perception of Color Visual Rendering:** Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates, Immersive Photos and Videos

UNIT– V:

Motion in Real and Virtual Worlds: Velocities and Accelerations, The Vestibular System, Physics in the Virtual World, Mismatched Motion and Vection

Interaction: Motor Programs and Remapping, Locomotion, Social Interaction

Audio: The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering

Text Books:

1. Augmented Reality: Principles & Practice by Schmalstieg / Hollerer, Pearson Education India;First edition (12 October 2016),ISBN-10: 9332578494
2. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016

Reference Books:

1. Allan Fowler-AR Game Development‖, 1st Edition, A press Publications, 2018, ISBN 978-1484236178
2. Understanding Virtual Reality: Interface, Application and Design, William R Sherman and Alan B Craig, (The Morgan Kaufmann Series in Computer Graphics)". Morgan Kaufmann Publishers, San Francisco, CA, 2002

3. Developing Virtual Reality Applications: Foundations of Effective Design, Alan B Craig, William R Sherman and Jeffrey D Will, Morgan Kaufmann, 2009
4. Designing for Mixed Reality, Kharis O'Connell Published by O'Reilly Media, Inc., 2016, ISBN:9781491962381
5. Sanni Siltanen- Theory and applications of marker-based augmented reality. Julkaisija – Utgivare Publisher. 2012. ISBN 978-951-38-7449-0
6. Gerard Jounghyun Kim, "Designing Virtual Systems: The Structured Approach", 2005

III Year II Semester	DEVOPS (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of this course are to:

1. Understand the principles of **DevOps**, its integration with Agile and SDLC, and its significance in modern software delivery.
2. Gain hands-on knowledge of **source code management** with Git and **UNIT-testing tools** like JUNIT-, NUNIT-, and SonarQube.
3. Learn the automation of **builds** and **Continuous Integration (CI)** using Jenkins and related tools.
4. Explore **Continuous Delivery (CD)** processes, **containerization using Docker**, and testing automation using tools like Selenium.
5. Implement **Configuration Management** with tools like **Ansible**, and understand **container orchestration** with **Kubernetes/OpenShift** along with Puppet and Chef.

Course Outcomes:

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

1. Explain the core concepts of DevOps, its lifecycle, architecture, and integration with Agile methodologies and software development workflows.
2. Use Git for version control and collaborate using Git commands, branching, and staging; apply UNIT- testing and analyze code quality using SonarQube.
3. Implement build automation and Continuous Integration using Jenkins pipelines, master-slave architecture, and manage Jenkins jobs and builds.
4. Demonstrate continuous delivery and deployment using Docker containers, and perform automated testing using tools like Selenium.
5. Apply configuration management using Ansible, and orchestrate containerized applications using Kubernetes/OpenShift; understand Puppet and Chef basics.

UNIT--I

Introduction to DevOps: Introduction to SDLC, Agile Model. Introduction to Devops. DevOps Features, DevOps Architecture, DevOps Lifecycle, Understanding Workflow and principles, Introduction to DevOps tools, Build Automation, Delivery Automation, Understanding Code Quality, Automation of CI/ CD. Release management, Scrum, Kanban, delivery pipeline, bottlenecks, examples

UNIT--II

Source Code Management (GIT): The need for source code control, The history of source code management, Roles and code, source code management system and migrations. What is Version Control and GIT, GIT Installation, GIT features, GIT workflow, working with remote repository, GIT commands, GIT branching, GIT staging and collaboration. UNIT- TESTING - CODE COVERAGE: JUNIT-, nUNIT- & Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

UNIT--III

Build Automation - Continuous Integration (CI): Build Automation, What is CI Why Cl is Required, CI tools, Introduction to Jenkins (With Architecture), jenkins workflow, jenkins master slave architecture, Jenkins Pipelines, PIPELINE BASICS - Jenkins Master, Node, Agent, and Executor Freestyle Projects & Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.

UNIT--IV

Continuous Delivery (CD): Importance of Continuous Delivery, CONTINUOUS DEPLOYMENT CD Flow, Containerization with Docker: Introduction to Docker, Docker installation, Docker commands, Images & Containers, DockerFile, Running containers, Working with containers and publish to Docker Hub.

Testing Tools: Introduction to Selenium and its features, JavaScript testing.

UNIT--V

Configuration Management - ANSIBLE: Introduction to Ansible, Ansible tasks, Roles, Jinja templating, Vaults, Deployments using Ansible.

CONTAINERIZATION USING KUBERNETES(OPENSHIFT): Introduction to Kubernetes Namespace & Resources, CI/CD - On OCP, BC, DC & ConfigMaps, Deploying Apps on Openshift Container Pods. Introduction to Puppet master and Chef.

Text Books:

1. Joyner, Joseph., Devops for Beginners: Devops Software Development Method Guide for Software Developers and It Professionals, 1st Edition Mihails Konoplows, 2015.
2. Alisson Machado de Menezes., Hands-on DevOps with Linux,1st Edition, BPB Publications, India, 2021.

Reference Books:

1. Len Bass, Ingo Weber, Liming Zhu. DevOps: A Software Architect's Perspective. Addison Wesley; ISBN-10
2. Gene Kim Je Humble, Patrick Debois, John Willis. The DevOps Handbook, 1st Edition, IT Revolution Press, 2016.
3. Verona, Joakim Practical DevOps, 1st Edition, Packt Publishing, 2016.

4. Joakim Verona. Practical Devops, Ingram short title; 2nd edition (2018). ISBN10: 1788392574
5. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications. ISBN: 9788126579952

III Year II Semester	GENERATIVE AI (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to:

1. Provide a **comprehensive understanding** of generative AI, its models, use cases, and ethical implications.
2. Explore **generative models for text** including transformers, attention mechanisms, GPT/BERT, and prompt engineering techniques.
3. Introduce **image generation** using GANs, VAEs, diffusion models, and transformer-based visual generation architectures.
4. Study **creative applications** of Gen AI in generating art, music, and autonomous agents using RNNs, GANs, and reinforcement learning.
5. Familiarize students with **open-source frameworks**, model fine-tuning, LLM programming, and deployment tools like Hugging Face and LangChain.

Course Outcomes:

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

1. Explain the fundamentals of generative AI, its historical development, key model types (GANs, VAEs, autoregressive, diffusion), and ethical implications in AI.
2. Analyze and apply generative text models such as GPT and BERT; demonstrate prompt engineering, transformer architecture, and tackle LLM-specific challenges.
3. Illustrate image generation using GANs, VAEs, Stable Diffusion, and transformer-based vision models like CLIP, DALL ·E, and GPT-4V.
4. Explore generative models for creative domains including painting, music, and games using neural style transfer, MuseGAN, and reinforcement learning.
5. Utilize and fine-tune open-source generative models (like LLaMA, GPT4All); implement solutions using frameworks like LangChain, Hugging Face, and deploy GenAI applications.

UNIT- I :

Introduction To Gen Ai: Historical Overview of Generative modelling, Difference between Gen AI and Discriminative Modeling, Importance of generative models in AI and Machine Learning, Types of Generative models, GANs, VAEs, autoregressive models and Vector quantized Diffusion models, Understanding of probabilistic modeling and generative process, Challenges of Generative Modeling, Future of Gen AI, Ethical Aspects of AI, Responsible AI, Use Cases.

UNIT- II:

Generative Models For Text: Language Models Basics, Building blocks of Language models, Transformer Architecture, Encoder and Decoder, Attention mechanisms, Generation of Text, Models like BERT and GPT models, Generation of Text, Autoencoding, Regression Models, Exploring ChatGPT, Prompt Engineering: Designing Prompts, Revising Prompts using Reinforcement Learning from Human Feedback (RLHF), Retrieval Augmented Generation, Multimodal LLM, Issues of LLM like hallucination.

UNIT- III:

Generation of Images: Introduction to Generative Adversarial Networks, Adversarial Training Process, Nash Equilibrium, Variational Autoencoders, Encoder-Decoder Architectures, Stable Diffusion Models, Introduction to Transformer-based Image Generation, CLIP, Visual Transformers ViT- Dall-E2 and Dall-E3, GPT-4V, Issues of Image Generation models like Mode Collapse and Stability.

UNIT- IV:

Generation of Painting, Music, and Play: Variants of GAN, Types of GAN, Cyclic GAN, Using Cyclic GAN to Generate Paintings, Neural Style Transfer, Style Transfer, Music Generating RNN, MuseGAN, Autonomous agents, Deep Q Algorithm, Actor-critic Network.

UNIT- V:

Open Source Models And Programming Frameworks: Training and Fine tuning of Generative models, GPT 4 All, Transfer learning and Pretrained models, Training vision models, Google Copilot, Programming LLM, LangChain, Open Source Models, Llama, Programming for TimeSformer, Deployment, Hugging Face.

Text Books:

1. Denis Rothman, “Transformers for Natural Language Processing and Computer Vision”, Third Edition , Packt Books, 2024

Reference Books:

1. David Foster, ”Generative Deep Learning”, O’Reily Books, 2024.
2. Altaf Rehmani, “Generative AI for Everyone”, BlueRose One, 2024.

III Year II Semester	INTELLIGENT AI AGENTS (Professional Elective-II)	L	T	P	C
		3	0	0	3

Course Objectives:

The primary objectives of this course are to:

1. Introduce the concept, characteristics, and classifications of intelligent agents and their interaction with environments.
2. Explore different **agent architectures** including reflex, goal-based, utility-based, and hybrid designs.
3. Understand foundational concepts of **knowledge representation and planning** using logic, ontologies, and classical planning methods.
4. Study various learning paradigms in agents, including **reinforcement learning**, Q-learning, and deep learning-based models.
5. Provide exposure to **real-world applications** of intelligent agents and multi-agent systems in robotics, games, IoT, and conversational AI.

Course Outcomes :

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

1. Explain the characteristics, types, and models of intelligent agents and differentiate between various agent environments.
2. Analyze and compare various agent architectures including reflex, goal-based, utility-based, and hybrid models.
3. Apply knowledge representation techniques and formulate planning problems using classical and probabilistic models.
4. Implement learning methods for agents using supervised, unsupervised, and reinforcement learning including Q-learning and policy gradients.
5. Demonstrate the use of intelligent agents in applications such as games, robotics, and IoT using simulation tools and assess ethical considerations.

UNIT-- I:

Introduction to Intelligent Agents-Definition and characteristics of intelligent agents, Types of agents: reactive, deliberative, hybrid, learning, PEAS (Performance measure, Environment, Actuators, Sensors) model, Agent environments: deterministic vs. stochastic, episodic vs. sequential, etc., Multi-agent systems: basic concepts

UNIT-- II:

Agent Architectures-Simple reflex agents, Model-based reflex agents, Goal-based and utility-based agents, Layered and hybrid architectures (Subsumption, InteRRaP, BDI), Behavior-based and reactive planning.

UNIT-- III:

Knowledge Representation and Planning-Propositional and predicate logic, Semantic networks and ontologies, Planning problem formulation, Classical planning: STRIPS, Graphplan, Partial-order planning, Planning under uncertainty (MDP, POMDP basics)

UNIT- IV:

Learning in Intelligent Agents-Supervised, unsupervised, and reinforcement learning, Q-learning and deep Q-networks (DQN),Policy gradient and actor-critic methods, Learning agent models (exploration vs. exploitation),Adaptation in dynamic environments

UNIT- V:

Applications and Tools-Game-playing agents, Agents in robotics and IoT, Chatbots and conversational agents, Simulation platforms: NetLogo, MESA, OpenAI Gym, UNIT-y ML-Agents, Ethical and societal implications of autonomous agents

Textbooks:

1. **Stuart Russell & Peter Norvig**, *Artificial Intelligence: A Modern Approach*, Pearson, 4th Edition, 2020.
2. **Michael Wooldridge**, *An Introduction to MultiAgent Systems*, Wiley, 2nd Edition, 2009.

Reference Books & Resources:

- **Nils J. Nilsson**, *The Quest for Artificial Intelligence*, Cambridge University Press.
- **Gerhard Weiss**, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, MIT Press.
- Research papers from *AAAI*, *IJCAI*, *AAMAS*, *NeurIPS*.
- Online courses: Coursera (Reinforcement Learning, Multi-Agent Systems), edX, Stanford CS221.

III Year II Semester	SOFTWARE PROJECT MANAGEMENT (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:

By the end of the course, the student shall be able to:

1. Understand the principles, practices, and evolution of software project management, including conventional and modern approaches.
2. Describe and analyze various life cycle phases and the associated artifacts produced during a software project.
3. Apply planning, estimation, and monitoring techniques to manage workflows, milestones, and iterations in software projects.
4. Evaluate different organizational structures, automation tools, and performance metrics for effective project control and team management.
5. Analyze and apply Agile and DevOps methodologies in real-world project environments to enhance flexibility, collaboration, and delivery speed.

Course Outcomes:

After completing this course, students will be able to:

1. Describe traditional and modern approaches in software project management, and analyze how improvements in software economics can be achieved.
2. Explain various life cycle phases of software projects and categorize the associated management and engineering artifacts.
3. Apply iterative planning principles and estimation techniques to define and manage project workflows, milestones, and schedules.
4. Evaluate different organizational structures, understand the role of process automation, and interpret project control metrics for better decision-making.
5. Demonstrate the application of Agile and DevOps practices in project environments, including deployment pipelines and team collaboration frameworks.

UNIT--I:

Conventional Software Management: The waterfall model, conventional software Management performance.

Evolution of Software Economics: Software Economics, pragmatic software cost estimation.

Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT--II:

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT-- III:

Model based software architectures: A Management perspective and technical perspective.

Work Flows of the process: Software process workflows, Iteration workflows.

Checkpoints of the process: Major mile stones, Minor Milestones, Periodic status assessments.

Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

UNIT-- IV:

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Process Automation: Automation Building blocks, The Project Environment.

Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

UNIT--V:

Agile Methodology, ADAPTing to Scrum, Patterns for Adopting Scrum, Iterating towards Agility. **Fundamentals of DevOps**: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system. DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

Text Books:

1. Software Project Management, Walker Royce, PEA, 2005.
2. Succeeding with Agile: Software Development Using Scrum, Mike Cohn, Addison Wesley.
3. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.

Reference Books:

1. Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH

2. Software Project Management, Joel Henry, PEA
3. Software Project Management in practice, Pankaj Jalote, PEA, 2005,
4. Effective Software Project Management, Robert K.Wysocki, Wiley,2006.
5. Project Management in IT, Kathy Schwalbe, Cengage

III Year II Semester	MOBILE ADHOC NETWORKS (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:

From this course, the student will learn:

1. How to architect sensor networks tailored to specific application scenarios.
2. Methods to devise suitable data dissemination protocols and efficiently model communication link costs.
3. The fundamental concepts of wireless sensor networks (WSNs) including architecture, protocols, and operational mechanisms at different layers.
4. How to evaluate the performance of sensor networks, identify performance bottlenecks, and propose solutions.
5. The security challenges associated with both Ad Hoc and wireless sensor networks and the mechanisms to mitigate them.

Course Outcomes:

After completing this course, the student will be able to:

1. Explain the characteristics, applications, and design challenges of Mobile Ad Hoc Networks (MANETs) and describe MAC layer protocols suited for such environments.
2. Classify and analyze routing and transport layer protocols in ad hoc networks and evaluate their suitability under varying network conditions.
3. Describe common security threats in ad hoc networks and apply secure routing protocols, intrusion detection mechanisms, and key management techniques.
4. Understand the structure and functioning of wireless sensor networks including sensor node design, communication range, clustering, and data retrieval mechanisms.
5. Apply knowledge of security protocols, sensor node operating systems, and simulation tools like NS-2 and TOSSIM for evaluating and prototyping WSN applications.

UNIT-- I:

Introduction to Ad Hoc Wireless Networks- Cellular and Ad Hoc Wireless Networks, Characteristics of MANETs, Applications of MANETs, Issues and Challenges of MANETs, Ad Hoc Wireless Internet, MAC protocols for Ad hoc Wireless Networks-Issues, Design Goals and Classifications of the MAC Protocols.

UNIT-- II: Routing Protocols for Ad Hoc Wireless Networks- Issues in Designing a Routing Protocol, Classifications of Routing Protocols, Topology-based versus Position-based Approaches, Issues and design goals of a Transport layer protocol, Classification of Transport layer solutions, TCP over Ad hoc Wireless Networks, Solutions for TCP over Ad Hoc Wireless Networks, Other Transport layer protocols.

UNIT-- III:

Security protocols for Ad hoc Wireless Networks- Security in Ad hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad hoc Wireless Networks, Cooperation in MANETs, Intrusion Detection Systems.

UNIT-- IV:

Basics of Wireless Sensors and Applications- The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications, Data Retrieval in Sensor Networks-Classification of WSNs, MAC layer, Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

UNIT-- V:

Security in WSNs- Security in WSNs, Key Management in WSNs, Secure Data Aggregation in WSNs, Sensor Network Hardware-Components of Sensor Mote, Sensor Network Operating Systems–TinyOS, LA-TinyOS, SOS, RETOS, Imperative Language-nesC, **Dataflow Style Language-**TinyGALS, Node-Level Simulators, NS-2 and its sensor network extension, TOSSIM.

Text Books:

1. Ad Hoc Wireless Networks – Architectures and Protocols, 1st edition, C. Siva Ram Murthy, B. S. Murthy, Pearson Education, 2004
2. Ad Hoc and Sensor Networks – Theory and Applications, 2nd edition *Carlos Corderio Dharma P.Aggarwal,* World Scientific Publications / Cambridge University Press, March 2006

Reference Books:

1. Wireless Sensor Networks: An Information Processing Approach, 1st edition, *Feng Zhao, Leonidas Guibas*, Elsevier Science imprint, Morgan Kauffman Publishers, 2005, rp2009
2. Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications, 1st edition, Subir Kumar Sarkar, et al., Auerbach Publications, Taylor & Francis Group, 2008
3. Ad hoc Networking, 1st edition,*Charles E.Perkins*, Pearson Education, 2001
4. Wireless Ad hoc Networking, 1st edition, *Shih-Lin Wu, Yu-Chee Tseng,* Auerbach Publications, Taylor & Francis Group, 2007

5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010

III Year II Semester	NATURAL LANGUAGE PROCESSING (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Provide foundational knowledge on Natural Language Processing and its real-world challenges.
2. Introduce language models, morphological analysis, and spelling correction techniques.
3. Equip students with methods for word-level analysis, including N-gram models and part-of-speech tagging.
4. Explore syntactic and semantic processing techniques, including parsing and meaning representation.
5. Examine discourse-level phenomena and introduce lexical resources and tools for advanced language understanding.

Course Outcomes:

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

1. Understand and apply core concepts of NLP including tokenization, morphology, and spelling correction.
2. Develop and evaluate statistical language models and use them in various NLP tasks.
3. Implement part-of-speech tagging using rule-based and probabilistic models.
4. Analyze and construct syntactic structures using context-free and dependency grammars.
5. Represent meaning through semantic frameworks and resolve word sense ambiguities.

UNIT-- I:

Introduction: Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance.

UNIT-- II:

Word Level Analysis: Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Part- of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models.

UNIT- -III:

Syntactic Analysis: Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification of feature structures

UNIT-- IV:

Semantics and Pragmatics: Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.

UNIT-- V:

Discourse Analysis and Lexical Resources: Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill's Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC).

Text Books:

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2ndEdition, Daniel Jurafsky, James H. Martin -Pearson Publication,2024.
2. Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, OReilly Media,2009.

Reference Books:

1. Language Processing with Java and Ling Pipe Cookbook, 1stEdition, Breck Baldwin, Atlantic Publisher, 2015.
2. Natural Language Processing with Java, 2ndEdition, Richard M Reese, OReilly Media,2015.
3. Handbook of Natural Language Processing, Second, NitinIndurkhya and Fred J. Damerau, Chapman and Hall/CRC Press, 2010.Edition
4. Natural Language Processing and Information Retrieval, 3rdEdition, TanveerSiddiqui, U.S. Tiwary, Oxford University Press,2008.

III Year II Semester	DISTRIBUTED OPERATING SYSTEM (Professional Elective-III)	L	T	P	C
		3	0	0	3

Course Objectives:

This course aims to:

1. Understand the fundamental principles and evolution of distributed computing systems and operating systems.
2. Learn about message passing mechanisms and inter-process communication in distributed environments.
3. Explore remote procedure calls and their implementation in distributed client-server models.
4. Analyze the design and implementation of Distributed Shared Memory (DSM) systems and synchronization mechanisms.
5. Examine resource management strategies and distributed file system design including fault tolerance and replication.

Course Outcomes :

After completing this course, the student will be able to:

1. Explain the architecture, models, and issues in distributed computing and operating systems.
2. Apply message passing techniques and remote procedure call mechanisms in distributed systems.
3. Analyze distributed shared memory structures, consistency models, and synchronization methods.
4. Evaluate resource management strategies including scheduling, process migration, and threading.
5. Demonstrate understanding of distributed file systems with respect to sharing, replication, caching, and fault tolerance.

UNIT-- I:

Fundamentals:

What is Distributed Computing Systems? Evolution of Distributed Computing System; Distributed Computing System Models; What is Distributed Operating System? Issues in Designing a Distributed Operating System; Introduction to Distributed Computing Environment (DCE).

Message Passing:

Introduction, Desirable features of a Good Message Passing System, Issues in PC by Message Passing, Synchronization, Buffering, Multi-datagram Messages, Encoding and

Decoding of Message Data, Process Addressing, Failure Handling, Group Communication, Case Study: 4.3 BSD UNIX IPC Mechanism.

UNIT-- II:

Remote Procedure Calls:

Introduction, The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Marshaling Arguments and Results, Server Management, Parameter-Passing Semantics, Call Semantics, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling, Security, Some Special Types of RPCs, RPC in Heterogeneous Environments, Lightweight RPC, Optimization for Better Performance, Case Studies: Sun RPC

UNIT--III:

Distributed Shared Memory:

Introduction, General Architecture of DSM systems, Design and Implementation Issues of DSM, Granularity, Structure of Shared Memory Space, Consistency Models, Replacement Strategy, Thrashing, Other approaches to DSM, Heterogeneous DSM, Advantages of DSM. Synchronization: Introduction, Clock Synchronization, Event Ordering, Mutual Exclusion, Dead Lock, Election Algorithms

UNIT-- IV:

Resource Management:

Introduction, Desirable Features of a Good Global Scheduling Algorithm, Task Assignment Approach, Load – Balancing Approach, Load – Sharing Approach Process Management: Introduction, Process Migration, Threads.

UNIT- -V:

Distributed File Systems:

Introduction, Desirable Features of a Good Distributed File System, File models, File– Accessing Models, File – Sharing Semantics, File – Caching Schemes, File Replication, Fault Tolerance, Atomic Transactions and Design Principles.

Text books:

1. **Pradeep K. Sinha**, *Distributed Operating Systems: Concepts and Design*, **PHI Learning, Latest Reprint (2023 or most recent)**, **ISBN: 978-8120313804**.
2. **George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair**, *Distributed Systems: Concepts and Design*, **Pearson Education, 5th Edition, Latest Indian Reprint (2020 or newer)**, **ISBN: 978-9353437350**.

Reference Books:

1. Andrew S. Tanenbaum: Distributed Operating Systems, Pearson Education, 2013.
2. Ajay D. Kshemkalyani and MukeshSinghal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 2008
3. SUNIT-aMahajan, Seema Shan, “ Distributed Computing”, Oxford University Press,2015

III Year II Semester	ADVANCED DATA STRUCTURES LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

- To gain practical experience in implementing and debugging advanced data structures.
- To develop skills in analyzing the time and space complexity of different implementations.
- To strengthen problem-solving abilities through competitive coding techniques.

List of Suggested Lab Experiments

1. Implementation of AVL Tree with insertion and deletion
2. Implementation of Red-Black Tree
3. Construction and operations on B-Trees and B+ Trees
4. Segment Tree and Fenwick Tree implementation for range queries
5. Binary Heap and Priority Queue operations
6. Union-Find operations with path compression
7. Open Addressing and Cuckoo Hashing
8. Trie and Suffix Trie implementation
9. Dijkstra's and Bellman-Ford shortest path algorithms
10. Kruskal's and Prim's algorithm for Minimum Spanning Tree
11. BFS and DFS traversal of graphs
12. Project: Design and implement a mini-application (e.g., autocomplete search, dynamic connectivity checker)

Note: Lab may be implemented in C/Java/Python as per Teacher preference.

III Year II Semester	MACHINE LEARNING LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

- To learn about computing central tendency measures and Data preprocessing techniques
- To learn about classification and regression algorithms
- To apply different clustering algorithms for a problem.

Software Required: Python/R/Weka

Lab should cover the concepts studied in the course work, sample list of Experiments:

1. Compute Central Tendency Measures: Mean, Median, Mode Measure of Dispersion: Variance, Standard Deviation.
2. Apply the following Pre-processing techniques for a given dataset.
   a. Attribute selection b. Handling Missing Values c. Discretization
   d. Elimination of Outliers
3. Apply KNN algorithm for classification and regression
4. Demonstrate decision tree algorithm for a classification problem and perform parameter tuning for better results
5. Demonstrate decision tree algorithm for a regression problem
6. Apply Random Forest algorithm for classification and regression
7. Demonstrate Naïve Bayes Classification algorithm.
8. Apply Support Vector algorithm for classification
9. Demonstrate simple linear regression algorithm for a regression problem
10. Apply Logistic regression algorithm for a classification problem
11. Demonstrate Multi-layer Perceptron algorithm for a classification problem
12. Implement the K-means algorithm and apply it to the data you selected. Evaluate performance by measuring the sum of the Euclidean distance of each example from its class center. Test the performance of the algorithm as a function of the parameters K.
13. Demonstrate the use of Fuzzy C-Means Clustering
14. Demonstrate the use of Expectation Maximization based clustering algorithm

III Year II Semester	SOFT SKILLS (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

Course Objectives:

This course aims to:

1. Develop students' analytical thinking, listening, and verbal/non-verbal communication skills.
2. Equip students with essential self-management strategies including stress and time management.
3. Improve grammatical and writing skills for professional and academic settings.
4. Prepare students for employment through training in group discussions, resume writing, and interview techniques.
5. Enhance interpersonal effectiveness and relationship management skills.

Course Outcomes (COs):

After successful completion of this course, students will be able to:

1. Demonstrate improved listening, analytical thinking, and communication skills through verbal and non-verbal expressions.
2. Apply self-management techniques such as stress, anger, and time management to handle personal and professional challenges effectively.
3. Use standard English grammar, writing, and note-making skills to communicate clearly in written formats.
4. Prepare job-oriented documents and perform confidently in group discussions and interviews.
5. Establish and maintain positive interpersonal relationships by understanding different styles and factors influencing them.

UNIT- – I:

Analytical Thinking & Listening Skills: Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.

Communication Skills: Verbal Communication; Non Verbal Communication (Body Language)

UNIT- – II:

Self-Management Skills: Anger Management, Stress Management, Time Management,

Six Thinking Hats, Team Building, Leadership Qualities
Etiquette: Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

UNIT- – III:
Standard Operation Methods : Basic Grammars, Tenses, Prepositions, Pronunciation, Letter Writing; Note Making, Note Taking, Minutes Preparation, Email & Letter Writing

UNIT--IV:
Job-Oriented Skills: Group Discussion, Mock Group Discussions, Resume Preparation, Interview Skills, Mock Interviews

UNIT--V:
Interpersonal relationships: Introduction, Importance, Types, Uses, Factors affecting interpersonal relationships, Accommodating different styles, Consequences of interpersonal relationships

Text books:

1. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.
2. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.

Reference books:

1. R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand& Company Ltd., 2018.
2. Raman, Meenakshi& Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.

E-resources:

1. https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_CAMBR_01

III Year II Semester	TECHNICAL PAPER WRITING & IPR	L	T	P	C
		2	0	0	-

Course Objectives:

1. To introduce the fundamentals of technical report writing, including sentence formation and tense usage.
2. To help students understand the principles of formatting and structuring technical documents.
3. To develop skills in proofreading, summarizing, and proposal writing.
4. To enable students to effectively use word processors for professional document preparation and presentation.
5. To create awareness about Intellectual Property Rights (IPR) and the process of patenting and innovation.

Course Outcomes :

After successful completion of the course, the student will be able to:

1. Understand the fundamentals of technical writing, sentence formation, tense usage, and report structure with clarity and precision.
2. Draft and refine technical documents using appropriate formatting, visuals, and plain English principles to enhance clarity and readability.
3. Apply proofreading techniques and summarization skills while preparing and presenting final technical reports and proposals effectively.
4. Utilize advanced word processing tools for organizing, editing, and securing technical documents professionally.
5. Understand the basic concepts of Intellectual Property Rights (IPR), including patents, copyrights, and international cooperation mechanisms.

UNIT-- I:

Introduction: An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.

Planning and Structuring: Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

UNIT- II:

Drafting report and design issues: The use of drafts, Illustrations and graphics.

Final edits: Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

UNIT- III:

Proofreading and summaries: Proofreading, summaries, Activities on summaries. **Presenting final reports:** Printed presentation, Verbal presentation skills, Introduction to proposals and practice.

UNIT- IV:

Using word processor:

Adding a Table of Contents, Updating the Table of Contents, Deleting the Table of Contents, Adding an Index, Creating an Outline, Adding Comments, Tracking Changes, Viewing Changes, Additions, and Comments, Accepting and Rejecting Changes , Working with Footnotes and Endnotes, Inserting citations and Bibliography, Comparing Documents, Combining Documents, Mark documents final and make them read only., Password protect Microsoft Word documents., Using Macros,

UNIT- V:

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

Text Books:

1. Kompal Bansal & Parshit Bansal, "Fundamentals of IPR for Beginner's", 1st Ed., BS Publications, 2016.
2. William S. Pfeiffer and Kaye A. Adkins, "Technical Communication: A Practical Approach", Pearson.
3. Ramappa,T., "Intellectual Property Rights Under WTO", 2nd Ed., S Chand, 2015.

Reference Books:

1. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011.
2. Day R, How to Write and Publish a Scientific Paper, Cambridge University Press(2006)

E-resources:

1. https://www.udemy.com/course/reportwriting/
2. https://www.udemy.com/course/professional-business-english-and-technical-report-writing/
3. https://www.udemy.com/course/betterbusinesswriting/

IV Year I Semester	INTERNET OF THINGS	L	T	P	C
		3	0	0	3

Course Objectives:

1. To understand the evolution, components, and enabling technologies of IoT.
2. To study sensing, actuation, and processing mechanisms in IoT systems.
3. To explore various IoT connectivity and communication protocols.
4. To examine interoperability standards and fog computing in IoT.
5. To analyze emerging IoT paradigms, challenges, and real-world applications.

Course Outcomes:

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

1. Explain the foundational concepts and technological evolution of IoT.
2. Identify and describe sensors, actuators, and processing strategies used in IoT devices.
3. Select suitable connectivity and communication technologies for specific IoT applications.
4. Apply interoperability frameworks and fog computing concepts to design IoT systems.
5. Analyze current trends, challenges, and apply IoT solutions to real-world domains like agriculture and transportation.

UNIT- I:

Predecessors of IoT: Introduction, Wireless Sensor Networks, Machine-to-Machine Communications, Cyber Physical Systems

Emergence of IoT: Introduction, Evolution of IoT, Enabling IoT and the Complex Interdependence of Technologies, IoT Networking Components, Addressing Strategies in IoT

UNIT- II:

IoT Sensing and Actuation: Introduction, Sensors, Sensor Characteristics, Sensorial Deviations, Sensing Types, Sensing Considerations, Actuators, Actuator Types, Actuator Characteristics

IoT Processing Topologies and Types: Data Format, Importance of Processing in IoT, Processing Topologies, IoT Device Design and Selection Considerations, Processing Offloading.

UNIT- III:

IoT Connectivity Technologies: Introduction, IEEE 802.15.4, Zigbee, Thread, ISA100.11A, WirelessHART, RFID, NFC,DASH7, Z-Wave, Weightless, Sigfox, LoRa, NB-IT, Wi-Fi, Bluetooth

IoT Communication Technologies: Introduction, Infrastructure Protocols, Discovery Protocols, Data Protocols, Identification Protocols, Device Management, Semantic Protocols.

UNIT- IV:

IoT Interoperability: Introduction, Standards, Frameworks

Fog Computing and Its Applications: Introduction, View of Fog Computing Architecture, Fog Computing in IoT, Selected Applications of Fog Computing

UNIT- V:

Paradigms, Challenges, and the Future: Introduction, Evolution of New IoT Paradigms, Challenges Associated with IoT, Emerging Pillars of IoT

IoT Case Studies: Agricultural IoT, Vehicular IoT

Text Books:

1. Introduction to IoT, Sudip Misra, Anandarup Mukhaerjee, Arjit Roy, Cambridge University Press, 2021
2. Internet of Things: Architecture, Design Principles and Applications, Rajkamal, McGraw Hill Higher Education

Reference Books:

1. Fog and Edge Computing: Principles and Paradigms, Rajkumar Buyya (Editor), Satish narayana Srirama (Editor) , ISBN: 978-1-119-52498-4, January 2019
2. Getting Started with the Internet of Things, CunoPfister , Oreilly

IV Year I Semester	HUMAN RESOURCES & PROJECT MANAGEMENT	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to

1. Provide knowledge about HR planning, recruitment, selection, and job design.
2. Develop skills in managing HR functions such as performance appraisal, compensation, and employee relations.
3. Emphasize the importance of ethical practices and HR audits in maintaining organizational health.
4. Understand the HRD framework and its impact on organizational success.
5. **Improve group interaction and team dynamics** for better collaboration and performance.
6. Understand the Fundamentals of Project Management and Project Networks
7. Implement appropriate management strategies tailored to specific challenges in different project types.

Course Outcomes:

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

1. **Explain** the principles and functions of Human Resource Management, including HR planning, recruitment, selection, and job design.
2. **Demonstrate** effective HR practices such as training & development, performance appraisal, compensation, and employee relations in organizational contexts.
3. **Apply** ethical standards, HR audit models, and Human Resource Development (HRD) frameworks to assess and improve organizational performance.
4. **Understand and utilize** project management fundamentals including project life cycle, planning, monitoring, and control for various types of projects.
5. **Evaluate** project implementation strategies, human factors, and review mechanisms for successful execution and closure of projects.

UNIT –I:

HRM: Nature, Scope, Concept of HRM, Functions of HRM, Role of HR manager, emerging trends in HRM, E-HRM, HR audit models, ethical aspects of HRM. HR Planning, Demand and Supply forecasting of HR, Job Design, Recruitment, Sources of recruitment, Selection- Selection Procedure.

UNIT –II:

HRD, HR accounting, Models, Concept of Training and Development, Methods of Training. Performance Appraisal: Importance Methods of performance appraisal, Career Development and Counseling, group interaction.

UNIT –III:

Basics of Project Management, Concept, resource management, Project environment, Types of Projects, project networks-DPR, Project life cycle, Project proposals, Monitoring project progress, Project appraisal and Project selection, 80-20 rules, production technology, communication matrix

UNIT-IV:

Identify various project types and their unique management challenges and apply appropriate management strategies for each. Project Implementation and Review: Forms of project organization, project planning, project control, human aspects of project management, prerequisites for successful project implementation, project review, performance evaluation, abandonment analysis

UNIT-V:

Project Implementation and Review: Forms of project organization, project planning, project control, human aspects of project management, prerequisites for successful project implementation, project review, performance evaluation, abandonment analysis

Text Books:

1. Robert L. Mathis, John H. Jackson, Manas Ranjan Tripathy, Human Resource Management, Cengage Learning 2016.
2. Sharon Pande and Swapnalekha Basak, Human Resource Management, Text and Cases, Vikas Publishing, 2e, 2016.
3. Stewart R. Clegg, Torgeir Skyttermoen, Anne Live Vaagaasar, Project Management, Sage Publications, 1e, 2021.
4. K. Nagarajan, Project Management, New Age International Publishers, 8e, 2017.

Reference Books :

1. Subba Rao P, “Personnel and Human Resource Management-Text and Cases”, Himalaya Publications, Mumbai, 2013.
2. K Aswathappa, “Human Resource and Personnel Management”, Tata McGraw Hill, New Delhi, 2013.
3. Prasanna Chandra, “Projects, Planning, Analysis, Selection, Financing, Implementation and Review”, Tata McGraw Hill Company Pvt. Ltd., New Delhi, 1998.
4. Vasanth Desai, “Project Management”, 4th edition, Himalaya Publications, 2018.
5. Lalitha Balakrishnan, Gowri, “Project Management”, Himalaya publishing house, New Delhi, 2022.

IV Year I Semester	SOFTWARE ARCHITECTURE & DESIGN PATTERNS (Professional Elective-IV)	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to:

1. Introduce the concept, types, and organization of **design patterns** in software development.
2. Explain the **object-oriented development paradigm**, its concepts, benefits, and drawbacks.
3. Provide insight into the **analysis and design phases** of object-oriented software development.
4. Explore and apply various **structural and architectural design patterns** including MVC.
5. Understand the design and implementation of **distributed object-oriented systems** and web services.

Course Outcomes:

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

1. **Define and classify** design patterns and explain their role in solving recurring design problems.
2. **Analyze and model** system requirements using object-oriented principles and domain knowledge.
3. **Apply** structural design patterns such as Adapter, Facade, and Proxy in software solutions.
4. **Design interactive systems** using the MVC architectural pattern and implement its components effectively.
5. **Develop distributed applications** using Java RMI, SOAP/RESTful web services, and integrate them using an Enterprise Service Bus.

UNIT- I

Introduction: What is a design pattern? Describing design patterns, the catalog of design pattern, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern What is object oriented development? key concepts of object oriented design other related concepts, benefits and drawbacks of the paradigm

UNIT – II

Analysis a System: Overview of the analysis phase, stage 1 gathering the requirements functional requirements specification, defining conceptual classes and relationships, using the knowledge of the domain Design and Implementation, discussions and further reading

UNIT – III

Design Pattern Catalog: Structural patterns, Adapter, bridge, composite, decorator, facade, flyweight, proxy.

UNIT – IV

Interactive systems and the MVC architecture: Introduction The MVC architectural pattern, analyzing a simple drawing program designing the system, designing of the subsystems, getting into implementation, implementing undo operation drawing incomplete items, adding a new feature pattern based solutions

UNIT – V

Designing with Distributed Objects: Client server system, java remote method invocation, implementing an object oriented system on the web, Web services (SOAP, Restful), Enterprise Service Bus

Text Books:

1. Object oriented analysis, design and implementation, brahma dathan, sarnath rammath , universities press,2013
2. Design patterns, Erich Gamma, Richard helan , Ralph johman , john vlissides, PEARSON Publication,2013

Reference Books:

1. Frank Bachmann, Regine Meunier , Hans Rohnert “Pattern Oriented Software Architecture” Volume 1, 1996.
2. William J Brown et al., "Anti Patterns: Refactoring Software, Architectures and Projects in Crisis", John Wiley, 1998

IV Year I Semester	DEEP LEARNING (Professional Elective-IV)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Provide foundational understanding of **machine learning evolution** and its transition into deep learning.
2. Introduce the principles of **artificial neural networks** and the training of deep networks.
3. Enable practical implementation of **deep learning models** using popular frameworks like Keras, TensorFlow, and PyTorch.
4. Explore advanced neural network architectures including **CNNs, RNNs, LSTMs, and Attention mechanisms**.
5. Examine real-world applications and emerging research areas such as **GANs, NLP, deep reinforcement learning**, and **autoencoders**.

Course Outcomes:

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

1. **Explain** the historical development and core principles of machine learning and deep learning.
2. **Design and train** artificial neural networks using suitable architectures for classification tasks.
3. **Implement and evaluate** deep learning models using Keras, TensorFlow, and PyTorch for various use cases.
4. **Apply** convolutional and recurrent neural networks to vision and sequence modeling problems.
5. **Analyze and experiment** with advanced deep learning applications like GANs, reinforcement learning, and generative models.

UNIT- I:

Fundamentals of Deep Learning: Artificial Intelligence, History of Machine learning: Probabilistic Modeling, Early Neural Networks, Kernel Methods, Random forests and Gradient Boosting Machines, **Fundamentals of Machine Learning:** Four Branches of Machine Learning, Evaluating Machine learning Models. **[Text Book 2]**

UNIT- II: Introducing Deep Learning: Biological and Machine Vision, Human and Machine Language, Artificial Neural Networks, Training Deep Networks, Improving Deep Networks. **[Text Book3]**

UNIT- III: Neural Networks: Anatomy of Neural Network, Introduction to Keras: Keras, TensorFlow, Theano and CNTK, Setting up Deep Learning Workstation, Classifying Movie Reviews: Binary Classification, Classifying newswires: Multiclass Classification. **[Text Book 2]**

UNIT- IV:

Convolutional Neural Networks: Nerual Network and Representation Learing, Convolutional Layers, Multichannel Convolution Operation, **Recurrent Neural Networks:** Introduction to RNN, RNN Code, PyTorch Tensors: Deep Learning with PyTorch, CNN in PyTorch. LSTM, Attention Mechanism**[Text Book 3]**

UNIT- V:

Interactive Applications of Deep Learning: Machine Vision, Natural Language processing, Generative Adversial Networks, Deep Reinforcement Learning. **[Text Book 1]**

Deep Learning Research: Autoencoders**,** Deep Generative Models: Boltzmann Machines Restricted Boltzmann Machines, Deep Belief Networks. **[Text Book 1]**

Text Books:

1. Deep Learning- Ian Goodfellow, Yoshua Bengio and Aaron Courvile, MIT Press, 2016
2. Deep Learning with Python - Francois Chollet, Released December 2017, Publisher(s): Manning Publications, ISBN: 9781617294433
3. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence - Jon Krohn, Grant Beyleveld, Aglaé Bassens, Released September 2019, Publisher(s): Addison-Wesley Professional, ISBN: 9780135116821
4. Deep Learning from Scratch - Seth Weidman, Released September 2019, Publisher(s): O'Reilly Media, Inc., ISBN: 9781492041412

Reference Books:

1. Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.
2. Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.
3. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education, 2004.

Web Link:

1. Swayam NPTEL: Deep Learning: https://onlinecourses.nptel.ac.in/noc22_cs22/preview

IV Year I Semester	COMPUTER VISION (Professional Elective-IV)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Introduce the fundamentals of **camera models**, **radiometry**, and **color representation** in imaging systems.
2. Develop an understanding of **image processing techniques** including filtering, edge detection, and texture analysis.
3. Explain the geometry of **multiple views**, **stereopsis**, and **image segmentation techniques**.
4. Familiarize students with **model fitting**, **tracking algorithms**, and **probabilistic methods** for inference.
5. Apply geometric camera models, calibration techniques, and model-based vision to **real-world vision applications**.

Course Outcomes:

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

1. **Describe** radiometry principles, light interaction with surfaces, and color models used in image formation.
2. **Apply** linear filters, Fourier transforms, edge detection, and texture analysis for image interpretation.
3. **Analyze** stereo vision systems and implement segmentation using clustering and graph-based methods.
4. **Use** probabilistic approaches and the EM algorithm for tracking and robust model fitting in dynamic systems.
5. **Calibrate** cameras using geometric models and solve vision problems in domains such as robotics and medical imaging.

UNIT –I:

Cameras: Pinhole Cameras Radiometry – Measuring Light: Light in Space, Light Surfaces, Important Special Cases Sources, Shadows, And Shading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, Interreflections: Global Shading Models Color: The Physics of Color, Human Color Perception, Representing Color, A Model for Image Color, Surface Color from Image Color.

UNIT-II:

Linear Filters: Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates, Edge **Detection**:Noise, Estimating Derivatives, Detecting Edges Texture0:Representing

Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape from Texture.

UNIT-III:

The Geometry of Multiple Views: Two Views Stereopsis: Reconstruction, Human Stereopsis, Binocular Fusion, Using More Cameras Segmentation by Clustering: What Is Segmentation?

Human Vision: Grouping and Getstalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

UNIT-IV:

Segmentation by Fitting a Model: The Hough Transform, Fitting Lines, Fitting Curves, Fitting as a Probabilistic Inference Problem, Robustness Segmentation and Fitting Using **Probabilistic Methods:** Missing Data Problems, Fitting, and Segmentation, The EM Algorithm in Practice,

Tracking With Linear Dynamic Models: Tracking as an Abstract Inference Problem, Linear Dynamic Models, Kalman Filtering, Data Association, Applications and Examples

UNIT- V:

Geometric Camera Models: Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection Equations

Geometric Camera Calibration: Least-Squares Parameter Estimation, A Linear Approach to Camera Calibration, Taking Radial Distortion into Account, Analytical Photogrammetry,

Text Books:

1. David A. Forsyth and Jean Ponce: Computer Vision – A Modern Approach, PHI Learning (Indian Edition), 2009.

Reference Books:

1. E. R. Davies: Computer and Machine Vision – Theory, Algorithms and Practicalities, Elsevier (Academic Press), 4th edition, 2013.
2. R. C. Gonzalez and R. E. Woods "Digital Image Processing" Addison Wesley 2008. 3. Richard Szeliski "Computer Vision: Algorithms and Applications" Springer-Verlag London Limited 2011.

IV Year I Semester	Blockchain Technology (Professional Elective-IV)	L	T	P	C
		3	0	0	3

Course Objectives:

1. To learn the fundamentals of Block Chain and various types of block chain and consensus mechanism.
2. To understand public block chain system, Private block chain system and consortium block chain.
3. Able to know the security issues of blockchain technology.

UNIT- – I:

Fundamentals of Blockchain: Introduction, Origin of Blockchain, Blockchain Solution, Components of Blockchain, Block in a Blockchain, The Technology and the Future.

Blockchain Types and Consensus Mechanism: Introduction, Decentralization and Distribution, Types of Blockchain, Consensus Protocol.

Cryptocurrency: Bitcoin, Altcoin and Token: Introduction, Bitcoin and the Cryptocurrency, Cryptocurrency Basics, Types of Cryptocurrencies, Cryptocurrency Usage.

UNIT- – II:

Public Blockchain System: Introduction, Public Blockchain, Popular Public Blockchains, The Bitcoin Blockchain, Ethereum Blockchain.

Smart Contracts: Introduction, Smart Contract, Characteristics of a Smart Contract, Types of Smart Contracts, Types of Oracles, Smart Contracts in Ethereum, Smart Contracts in Industry.

UNIT- – III:

Private Blockchain System: Introduction, Key Characteristics of Private Blockchain, Private Blockchain, Private Blockchain Examples, Private Blockchain and Open Source, E- commerce Site Example, Various Commands (Instructions) in E-commerce Blockchain, Smart Contract in Private Environment, State Machine, Different Algorithms of Permissioned Blockchain, Byzantine Fault, Multichain.

Consortium Blockchain: Introduction, Key Characteristics of Consortium Blockchain, Need of Consortium Blockchain, Hyperledger Platform, Overview of Ripple, Overview of Corda.

Initial Coin Offering: Introduction, Blockchain Fundraising Methods, Launching an ICO, Investing in an ICO, Pros and Cons of Initial Coin Offering, Successful Initial Coin Offerings, Evolution of ICO, ICO Platforms.

UNIT- – IV:

Security in Blockchain: Introduction, Security Aspects in Bitcoin, Security and Privacy Challenges of Blockchain in General, Performance and Scalability, Identity Management and Authentication, Regulatory Compliance and Assurance, Safeguarding Blockchain Smart Contract (DApp), Security Aspects in Hyperledger Fabric.

Applications of Blockchain: Introduction, Blockchain in Banking and Finance, Blockchain in Education, Blockchain in Energy, Blockchain in Healthcare, Blockchain in Real-estate, Blockchain in Supply Chain, The Blockchain and IoT. Limitations and Challenges of Blockchain.

UNIT- – V:

Blockchain Case Studies:
Case Study 1 – Retail,
Case Study 2 – Banking and Financial Services,
Case Study 3 – Healthcare,
Case Study 4 – Energy and Utilities.

Text book:

1. "Block chain Technology", Chandramouli Subramanian, Asha A.George, Abhilasj K A and Meena Karthikeyan , Universities Press.

Reference Books:

1. Blockchain Blue print for Economy, Melanie Swan, SPD Oreilly.
2. Blockchain for Business, Jai Singh Arun, Jerry Cuomo, Nitin Gauar, Pearson Addition Wesley

IV Year I Semester	AGILE METHODOLOGIES (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

The course aims to:

1. Introduce the fundamentals of **Agile values, principles**, and the Agile Manifesto.
2. Explain how Agile principles are applied to **project planning, delivery**, and **team collaboration**.
3. Provide practical understanding of **Scrum framework** and its core roles, artifacts, and events.
4. Explore **Extreme Programming (XP)** practices, feedback loops, and engineering excellence.
5. Illustrate the use of **Lean thinking, Kanban**, and the role of **Agile coaching** in facilitating organizational change.

Course Outcomes:

After successful completion of this course, students will be able to:

1. **Explain** Agile fundamentals, values, principles, and the rationale behind the Agile Manifesto.
2. **Apply** Agile practices in real-world projects for effective collaboration, execution, and continuous improvement.
3. **Demonstrate** the use of Scrum roles, events, and artifacts for iterative project management.
4. **Implement** XP practices such as refactoring, continuous integration, and incremental design to enhance code quality.
5. **Analyze** Lean and Kanban approaches to optimize workflow and facilitate Agile transformation through coaching.

UNIT I:

Introduction to Agile-Learning Agile: Getting Agile into Your Brain, Understanding Agile Values and Principles, No Silver Bullet – Agile to the Rescue, Adding Agile Makes a Difference, Fractured Perspectives and Project Failures, The Agile Manifesto and Purpose Behind Practices, Individuals and Interactions over Processes and Tools, Working Software over Comprehensive Documentation, Customer Collaboration over Contract Negotiation, Responding to Change over Following a Plan, Principles over Practices, Understanding Methodologies and Where to Start

UNIT II:

Agile Principles and Project Delivery-The 12 Agile Principles, Customer Collaboration and Clarifying Expectations, Agile in Action: eBook Reader Project Case Study, Communication and Team Coordination, Effective Project Execution and Work Environment, Continuous Improvement for Team and Product, Integrating Agile Principles in the Project Lifecycle

UNIT III:

Scrum and Agile Project Planning-Scrum Fundamentals and Roles, Scrum Master, Product Owner, Development Team, Scrum Values and Events, Daily Scrum, Feedback Loops, Retrospectives, Sprint Planning and Execution, Self-Organizing Teams and Collective Ownership, User Stories, Velocity, Story Points, Burndown Charts, Sprint Boards and Task Management, Cultural Compatibility and Scrum Success

UNIT IV:

Extreme Programming (XP) and Design Practices-XP Practices: Programming, Planning, Integration, Team, Embracing Change through XP Values and Mindset, XP Principles and Feedback Loops, Simplicity and Incremental Design, Identifying Code Smells and Refactoring, Continuous Integration and Avoiding Technical Debt, Collaborative Teams and Holistic Practices

UNIT V:

Lean, Kanban, and Agile Coaching-Lean Thinking and Eliminating Waste, Value Stream Mapping and Root Cause Analysis, Set-Based Development and Flow Efficiency, Delivering Fast and Managing Bottlenecks, Kanban Principles and Visualizing Workflow, Limiting Work in Progress and Measuring Flow, Agile Coaching Principles and Change Facilitation

Textbook:

1. Andrew Stellman & Jill Alison Hart, *Learning Agile*, O'Reilly, 2015

Reference Books:

1. Andrew Stellman & Jennifer Greene, *Head First Agile*, O'Reilly, 2017,
2. Rubin K., *Essential Scrum: A Practical Guide to the Most Popular Agile Process*, Addison-Wesley, 2013

IV Year I Semester	BIG DATA ANALYTICS (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Introduce the concept of **Big Data**, its real-world applications across industries, and the enabling technologies like Hadoop and cloud platforms.
2. Explain the fundamentals of **NoSQL databases**, their data models, replication strategies, and use of distributed storage systems like Cassandra.
3. Provide hands-on knowledge in **Hadoop architecture**, HDFS, MapReduce programming, and query processing with Hive.
4. Explore the **Apache Spark framework** for in-memory data processing, transformations, and data analytics.
5. Understand **real-time data processing** using Spark Streaming, structured streaming concepts, and performance tuning strategies.

Course Outcomes:

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

1. **Identify and analyze** Big Data applications and technologies used in various domains like healthcare, finance, and marketing.
2. **Design and implement** NoSQL data models using key-value stores, document stores, and graph databases with tools like Cassandra.
3. **Develop** Big Data processing solutions using Hadoop components such as HDFS, MapReduce, and Hive for structured and semi-structured data.
4. **Utilize** Apache Spark for high-performance, distributed in-memory computing with RDDs, DataFrames, and advanced transformations.
5. **Apply** stream processing concepts in Spark for handling real-time data, including event-time processing, windowing, and watermarks.

UNIT - I:

Big data, convergence of key trends, unstructured data, industry examples, web analytics, marketing, fraud detection, risk analysis, credit risk management, algorithmic trading, healthcare, medicine, advertising, Hadoop, open source technologies, cloud computing, mobile business intelligence, crowdsourcing analytics, inter and trans firewall analytics

UNIT- II:

NoSQL, aggregate data models, key-value data model, document data model, relationships, graph databases, schemaless databases, materialized views, distribution models, sharding, master-slave replication, peer-to-peer replication, consistency, version stamps, Cassandra, table creation, data loading, data reading

UNIT- III:

data formats, Hadoop, data analysis, scaling out, HDFS architecture, fault tolerance, data replication, high availability, data locality, MapReduce architecture, process flow, Java interface, data flow, Hadoop I/O, data integrity, compression, serialization, Hive, data types, file formats, HiveQL DDL, HiveQL DML, logical joins, window functions, optimization, table partitioning, bucketing, indexing, join strategies

UNIT- IV:

Apache Spark, advantages over Hadoop, lazy evaluation, in-memory processing, DAG, SparkContext, SparkSession, RDD, narrow transformations, wide transformations, actions, DataFrames, RDD to DataFrame conversion, Catalyst optimizer, transformations on DataFrames, working with dates, working with timestamps, handling nulls, complex types, JSON, grouping, window functions, joins, data sources, broadcast variables, accumulators, on-premises deployments, cluster managers, standalone mode, Spark on YARN, Spark logs, Spark UI, Spark UI history server, debugging, Spark first aid

UNIT- V:

Spark performance tuning, stream processing, event-time processing, stateful processing, tumbling windows, late data handling, watermarks, duplicate removal, structured streaming, streaming core concepts, structured streaming in action, stream transformations, stream input, stream output

Text Books:

1. Big Data, Big Analytics: Emerging, Michael Minnelli, Michelle Chambers, and AmbigaDhiraj, 1st edition ,2013
2. SPARK: The Definitive Guide, Bill Chambers &MateiZaharia, O'Reilley, 2018-first Edition.
3. Business Intelligence and Analytic Trends for Today's Businesses", Wiley, First edition-2013.
4. P. J. Sadalage and M. Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World Polyglot Persistence", Addison-Wesley Professional, 2012
5. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012

Reference Books:

1. "Hadoop Operations", O'Reilley, Eric Sammer, First Edition -2012.
2. "Programming Hive", O'Reilley, E. Capriolo, D. Wampler, and J. Rutherglen, 2012.
3. "HBase: The Definitive Guide", O'Reilley, Lars George, September 2011: First Edition..
4. "Cassandra: The Definitive Guide", O'Reilley, Eben Hewitt, 2010.
5. "Programming Pig", O'Reilley, Alan Gates, October 2011: First Edition

IV Year I Semester	MOBILE COMPUTING (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to:

1. Introduce the fundamentals of **mobile communication technologies**, their standards, and architecture.
2. Explain **GSM, GPRS, CDMA, 3G, and 4G architectures** and protocols used in mobile systems.
3. Provide insight into **mobile IP**, **ad hoc networks**, and **wireless sensor networks**.
4. Explore **data synchronization** and **mobile agent-based computing** for mobile environments.
5. Study **short-range wireless networks**, **WLAN**, **WAP**, and **mobile internet protocols**.

Course Outcomes:

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

1. **Describe** mobile communication techniques, architectures, and limitations of mobile systems.
2. **Compare and contrast** various mobile communication standards including GSM, CDMA, 3G, and 4G.
3. **Analyze** mobile network layer functionalities, mobile IP, MANETs, and wireless sensor networks.
4. **Apply** synchronization methods and mobile agent principles in mobile application development.
5. **Evaluate** WLAN architecture, WAP protocol stack, and mobile internet communication mechanisms.

UNIT--I:

Mobile Communications: An Overview- Mobile Communication-guided transmission, unguided transmission- signal propagation frequencies, antennae, modulation, modulation methods and standards for voice-oriented data communication standards, modulation methods and standards for data and voice communication, mobile computing- novel applications and limitations, mobile computing architecture, mobile system networks. Mobile devices and systems: Cellular networks and frequency reuse, Mobile smart phones, Smart mobiles and systems, handheld pocket computers, Handheld devices, Smart systems, Limitations of mobile devices

UNIT--II:

GSM and other 2G Architectures: GSM-services and system architecture, Radio interfaces of GSM, Protocols of GSM, Localization, Call handling, GPRS system architecture. Wireless medium access control, CDMA, 3G,and 4G

Communication: Modulation, Multiplexing, Controlling the medium access, Spread spectrum, Coding methods, IMT-20003G wireless communication standards, WCDMA 3G communication standards, CDMA 3G communication standards, Broadband wireless access, 4Gnetworks.

UNIT--III:

Mobile IP Network layer: IP and Mobile IP network layers: OSI layer functions, TCP/IP and Internet protocol, Mobile internet protocol; Packet delivery and Handover Management;

Location Management: Agent Discovery; Mobile TCP

Introduction to Mobile Adhoc network: fixed infrastructure architecture, MANET infrastructure architecture;

MANET: properties, spectrum, applications; Security in Ad-hoc network; Wireless sensor networks; sensor network applications.

UNIT--IV:

Synchronization: Synchronization in mobile computing systems, Usage models for Synchronization in mobile application, Domain-dependant specific rules for data synchronization, Personal information manager, synchronization and conflict resolution strategies, synchronizer; Mobile agent: mobile agent design, aglets; Application Server

UNIT--V:

Mobile Wireless Short Range Networks and Mobile Internet: Wireless networking and wireless LAN, Wireless LAN (WLAN) architecture, IEEE 802.11protocol layers, Wireless application protocol (WAP)-WAP1.1 architecture, wireless datagram protocol (WDP), Wireless Transport Layer Security (WTLS), wireless transaction and session layers, wireless application environment.

TEXTBOOK:

1. RAJ KAMAL, “Mobile Computing,” second edition, Oxford.
2. ASOKE K TALUKDER, HASANAHMED, ROOPA R YAVAGAL, “Mobile Computing, Technology Applications and Service Creation” Second Edition, Mc Graw Hill.
3. UWE Hansmann, Lother Merk, Martin S. Nocklous, Thomas Stober, “Principles of Mobile Computing,” Second Edition, Springer

IV Year I Semester	CYBER PHYSICAL SYSTEMS (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives:

This course is designed to:

1. Introduce **symbolic synthesis techniques** and **symbolic model construction** for designing cyber-physical systems (CPS).
2. Explain the **security requirements**, attack models, and countermeasures for securing CPS.
3. Familiarize students with **synchronization and consensus algorithms** in distributed CPS architectures.
4. Provide insights into **real-time scheduling techniques** including multiprocessor and memory-aware scheduling for CPS.
5. Teach students about **model integration** and formal methods for semantics and language integration in CPS modeling.

Course Outcomes:

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

1. **Construct symbolic models** for cyber-physical systems using synthesis techniques and relevant software tools.
2. **Analyze and design secure CPS architectures** by identifying security threats and proposing system-theoretic countermeasures.
3. **Apply synchronization and consensus algorithms** in distributed CPS and explain the role of formal software engineering.
4. **Implement real-time scheduling algorithms** that accommodate timing variability and resource constraints in multicore environments.
5. **Integrate models and semantics** using domain-specific modeling languages (DSMLs) and formal language integration methods for CPS design.

UNIT- I:

Symbolic Synthesis for Cyber-Physical Systems: Introduction and Motivation, Basic Techniques - Preliminaries, Problem Definition, Solving the Synthesis Problem, Construction of Symbolic Models, Advanced Techniques: Construction of Symbolic Models, Continuous-Time Controllers, Software Tools

UNIT- II:

Security of Cyber-Physical Systems: Introduction and Motivation, Basic Techniques - Cyber Security Requirements, Attack Model, Countermeasures, Advanced Techniques: System Theoretic Approaches

UNIT- III:

Synchronization in Distributed Cyber-Physical Systems: Challenges in Cyber-Physical Systems, A Complexity-Reducing Technique for Synchronization, Formal Software Engineering, Distributed Consensus Algorithms, Synchronous Lockstep Executions, Time-Triggered Architecture, Related Technology, Advanced Techniques

UNIT- IV:

Real-Time Scheduling for Cyber-Physical Systems: Introduction and Motivation, Basic Techniques - Scheduling with Fixed Timing Parameters, Memory Effects, Multiprocessor/Multicore Scheduling, Accommodating Variability and Uncertainty

UNIT- V:

Model Integration in Cyber-Physical Systems: Introduction and Motivation, Causality, Semantic Domains for Time, Interaction Models for Computational Processes, Semantics of CPS DSMLs, Advanced Techniques, ForSpec, The Syntax of CyPhyML, Formalization of Semantics, Formalization of Language Integration.

Text Books:

1. Raj Rajkumar, Dionisio De Niz, and Mark Klein, Cyber-Physical Systems, Addison-Wesley Professional, 2016
2. Rajeev Alur, Principles of Cyber-Physical Systems, MIT Press

Reference Books:

1. E.A.Lee, Sanjit Seshia, Introduction to Embedded Systems: A Cyber-Physical Systems Approach, MIT Press
2. Andre Platzer, Logical Foundations of Cyber-Physical Systems, (2e), Springer Publishing, 2018

IV Year I Semester	CLOUD COMPUTING (Professional Elective-V)	L	T	P	C
		3	0	0	3

Course Objectives

This course aims to:

1. Introduce the **fundamentals and service models** of cloud computing and its **deployment architectures**.
2. Explain the **enabling technologies** such as distributed computing, virtualization, and service-oriented architectures.
3. Familiarize students with **virtual machines, containers, and orchestration platforms** like Docker and Kubernetes.
4. Discuss **key challenges** in cloud computing including **security, scalability, interoperability, and energy efficiency**.
5. Expose learners to **advanced cloud topics** including serverless computing, IoT-cloud integration, edge/fog computing, and DevOps.

Course Outcomes

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

1. **Describe the architecture and service models** of cloud computing and explain the benefits and challenges of different deployment models.
2. **Apply distributed and parallel computing concepts** and demonstrate understanding of enabling technologies such as RPC, SOA, and Web services.
3. **Analyze and differentiate between virtualization and containerization technologies**, and implement orchestration using Kubernetes or Docker Swarm.
4. **Evaluate cloud-specific issues** including interoperability, scalability, fault tolerance, and security architectures.
5. **Explore and apply advanced concepts** like serverless computing, IoT-cloud interaction, edge/fog computing, and DevOps practices in cloud environments.

UNIT -I:

Introduction to Cloud Computing Fundamentals-Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).

UNIT-II:

Cloud Enabling Technologies-Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, virtualization.

UNIT-III:

Virtualization and Containers-Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

UNIT-IV:

Cloud computing challenges -Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

UNIT -V:

Advanced concepts in cloud computing-Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

Text Books:

1. Mastering Cloud Computing, 2nd edition, Rajkumar Buyya, Christian Vecchiola, Thamarai Selvi, Shivananda Poojara, Satish N. Srirama, Mc Graw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.

Reference Books:

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd edition, MK Elsevier, 2018.
2. Essentials of cloud Computing, K. Chandrasekhran, CRC press, 2014.
3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP)

IV Year I Semester	PROMPT ENGINEERING (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

Course Objectives:

The main objectives of the course are to

i. Apply iterative prompting for clarity and context.
ii. Create varied prompts to steer model outputs.
iii. Construct chain-of-thought and structured prompts.
iv. Develop retrieval-augmented pipelines to ground outputs.
v. Evaluate LLM agents and multimodal apps for ethics and robustness.

Unit I: Foundations of Prompt Engineering: Definition of prompt engineering, Distinction between prompt engineering and model fine-tuning, Motivation and benefits of prompt engineering, Core principles of effective prompt design**,** Anatomy of a prompt, Setting up the Python environment for LLM interaction, Iterative prompting lifecycle, Common prompt pitfalls and remediation

Lab Experiments:

1. Environment & Connectivity: Install required packages (e.g., transformers, openai); securely configure the API key; run a simple "Hello, world" prompt to verify model access.
2. Baseline vs. Enhanced Prompts: Execute a naïve prompt ("Write a one-paragraph bio of Ada Lovelace.") and an enhanced prompt that adds role framing, specificity, and explicit format instructions; compare both outputs for relevance, completeness, and style.
3. Iterative Refinement on a Simple Task: Summarize the plot of the Shakespearean play Romeo and Juliet in two sentences through three rounds of prompt tweaking:
   a. Minimal instruction.
   b. Addition of length and style constraints
   c. Specification of key content elements (setting and theme)

   Document how each iteration changes and improves the result.
4. Diagnosing Prompt Failures & Edge Cases: Craft a vague or contradictory prompt; analyze the failure mode (ambiguity, missing context, or format errors); refine the prompt by adding examples or clarifying instructions.

Unit II: Advanced Prompt Patterns & Techniques: Enhanced prompt anatomy: contextual detail and explicit output specifications, Few-shot in-context prompting, Prompt structuring and template design, Role-based prompting to establish personas or system behavior, Negative prompting to filter or suppress undesired content, Constraint specification and instruction enforcement (e.g., length, format), Iterative prompt refinement and optimization

Lab Experiments:

1. Few-Shot vs. Zero-Shot Comparison: Design and execute a zero-shot prompt and a few-shot prompt (with 2–3 exemplar input-output pairs) for a chosen text task (e.g., sentiment classification or translation); compare outputs for accuracy, consistency, and adherence to examples.
2. Role-Based & Negative Prompting: Craft a role-based prompt to establish a specific persona (e.g., "You are a financial advisor…"); then create a negative prompt to suppress undesired content (e.g., "Do not mention any brand names"); evaluate how each influences the model's response.
3. Constraint Specification & Iterative Refinement: Select an open-ended task (e.g., summarizing a technical article); issue a basic prompt; identify failures in length or format; refine the prompt by adding explicit constraints (word count, bullet format, etc.); document improvements over two refinement cycles.

Unit III: Structured Output & Reasoning Techniques: Importance of structured outputs for real-world applications, Prompting for specific formats (lists, tables, Markdown), Generating valid JSON and YAML via explicit instructions, Eliciting chain-of-thought reasoning in zero-shot prompts, Decomposing complex tasks into manageable sub-tasks

Lab Experiments:

1. Structured Format Prompting: Instruct the model to output information as bullet lists and Markdown tables (e.g., "List three benefits of daily exercise in a Markdown table with columns 'Benefit' and 'Description.'"); verify the output matches the requested structure.
2. JSON/YAML Generation: Provide a brief dataset description (e.g., three books with title, author, publication year) and prompt the model to produce valid JSON or YAML; use a parser to validate syntax and refine the prompt if errors occur.
3. Chain-of-Thought & Task Decomposition: Present a multi-step problem (e.g., a logic puzzle) and apply zero-shot CoT prompting (e.g., "Let's think step by step. Explain your reasoning before the final answer."); separately, decompose the problem into sequential sub-questions, collect partial answers, combine them, and compare accuracy against a direct-answer baseline.

Unit IV: Retrieval-Augmented Generation & LangChain Workflows: Limitations of LLM internal knowledge, Need for external data sources, Introduction to Retrieval-Augmented Generation (RAG), Overview of RAG architecture (indexing vs. retrieval + generation), Getting started with LangChain for LLM applications, Basics of LangChain Expression Language (LCEL), Simplified indexing pipeline: document loading & text splitting, Fundamentals of embeddings and vector stores, Building a basic retrieval-generation pipeline with an LCEL chain

Lab Experiments:

1. Building a Simple LCEL Chain: Create a minimal LCEL script that accepts a fixed instruction (e.g., "Summarize this text: …"), passes it to an LLM, and prints the result; verify end-to-end execution.
2. Basic Data Indexing for RAG: Load a small collection of documents; split into uniform chunks (e.g., 200 tokens); generate embeddings for each chunk; store them in an in-memory vector store; inspect for consistency.
3. Constructing & Running a Basic RAG Chain: Build a pipeline that:
   a. Receives a user query
   b. Retrieves the top-k relevant chunks
   c. Constructs a combined prompt with context + query
   d. Send it to the LLM
   e. Returns the answer

Test with sample queries and compare factual accuracy against a prompt without retrieval.

Unit V: Agents, Multimodal AI & Ethical Evaluation: Introduction to LLM agents and their basic architecture, Overview of multimodal AI models (VLMs), Prompting for text-to-image generation and image understanding, Importance of prompt evaluation beyond subjective judgment, Manual evaluation techniques (heuristic checks for accuracy, relevance, format), Introduction to "LLM-as-Judge" for automated evaluation, Security considerations (prompt injection, sensitive-information risks), Prompt-based mitigation strategies for safety and robustness, Ethical concerns (bias, misinformation, data privacy), Brief exploration of UI frameworks (Streamlit/Gradio) for deploying prompt-driven apps, Adapting to the evolving nature of prompt engineering through continuous learning

Lab Experiments:

1. Building a Simple LLM Agent: Register a tool (e.g., a calculator function) and craft prompts that instruct the agent to invoke it when required; implement using Lang Chain or a function-calling API; test on queries requiring tool execution.
2. Multimodal Prompting Exploration: Generate images from detailed text prompts; feed one generated image into an image-understanding model or API with an appropriate prompt; compare the returned caption to the original prompt to evaluate alignment.
3. Prompt Evaluation & Ethics Workshop:
   a. Select two existing prompts and generate multiple outputs; apply manual heuristic checks for accuracy, relevance, and format compliance.
   b. Use an "LLM-as-Judge" prompt (e.g., "Rate these outputs on a scale of 1–5 for clarity and correctness.") to automate evaluation.
   c. Design a prompt- injection test (e.g., "Ignore previous instructions…"), observe the response, then refine system prompts to mitigate the vulnerability.

IV Year I Semester	CONSTITUTION OF INDIA	L	T	P	C
		2	0	0	-

Course Objectives:

- Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
- To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
- To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

UNIT–I:

History of Making of the Indian Constitution: History, Drafting Committee, (Composition & Working)
Philosophy of the Indian Constitution- Preamble, Salient, Features

UNIT–II:

Contours of Constitutional Rights & Duties: Fundamental Rights, Right to Equality, Right to Freedom, Right against Exploitation, Right to Freedom of Religion, Cultural and Educational Rights, Right to Constitutional Remedies, Directive Principles of State Policy, Fundamental Duties.

UNIT–III:

Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, **Executive-** President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions

UNIT–IV:

Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative CEO of Municipal Corporation, Pachayati raj: Introduction, PRI: ZilaPachayat, Elected officials and their roles, CEO ZilaPachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

UNIT–V:

Election Commission: Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women.

Text Books:

1. The Constitution of India, 1st Edition, (Bare Act), Government Publication, 1950
2. Framing of Indian Constitution, 1st Edition, Dr. S. N. Busi, Dr. B. R. Ambedkar 2015

Reference Books:

1. Indian Constitution Law, 7th Edition, M. P. Jain, Lexis Nexis, 2014

JNTUGV- COLLEGE OF ENGINEERING VIZIANAGARAM

DEPARTMENT OF INFORMATION TECHNOLOGY

B.TECH- IT (R23)-HONORS

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA VIZIANAGARAM
VIZIANAGARAM – 535 003, Andhra Pradesh, India
COLLEGE OF ENGINEERING VIZIANAGARAM(A)

B.Tech IT (R23-COURSE STRUCTURE)

HONOR DEGREE IN INFORMATION TECHNOLOGY

(I) Cyber Security

S.No	Subject Title	L	T	P	C
1	Cyber Security Essentials	3	0	0	3
2	Secure Coding	3	0	0	3
3	Vulnerability Assessment & Penetration Testing	3	0	0	3
4	Malware Analysis	3	0	0	3
5	03 MOOCS courses @ 2credits each (8 weeks course) 02 MOOCS courses @ 3credits each (12 weeks course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(II) AI/ML

S.No	Subject Title	L	T	P	C
1	Statistical Machine Learning	3	0	0	3
2	Reinforcement Learning	3	0	0	3
3	Programming Foundations with GenAI Tools	3	0	0	3
4	Advanced Computer Vision	3	0	0	3
5	03 MOOCS courses @ 2credits each (8 weeks course) 02 MOOCS courses @ 3credits each (12 weeks course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(III) Data Science

S.No	Subject Title	L	T	P	C
1	Mathematical Essential for Data Science	3	0	0	3
2	Introduction to Data Science	2	0	2	3
3	Data Analytics and Visualization	2	0	2	3
4	Graph Analytics	2	0	2	3
5	03 MOOCS courses @ 2credits each (8 weeks course) 02 MOOCS courses @ 3credits each (12 weeks course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

Note: Students who have registered for honors program can opt a maximum of two (02) courses per semester.

		L	T	P	C
	Honor Course	3	1	0	4
CYBER SECURITY ESSENTIALS					

Course Objective:

1. To introduce information security concepts to undergraduate engineering students, so they can defend their personal and organizational information from probable security attacks and incidents.
2. Classify and analyze various categories of network attacks
3. Describe and compare various cryptographic algorithms

Course Outcomes:

By the end of this course, the student will be able to

1. Understand the basics and need for information security
2. Identify, analyze, and evaluate infrastructure and network vulnerabilities.
3. Understand and analyze different access control and authentication methods.
4. Identify and assess current and anticipated security risks and vulnerabilities with vulnerability assessment and auditing methods.
5. Learn the fundamentals of cryptography and how cryptography serves as the central language of information security.

UNIT-I:

Introduction to Security: Challenges of Securing Information, Definition of Information Security, Attackers, Attacks and Defenses.

Systems Threats and Risks: Software-Based Attacks, Hardware-Based Attacks, Attacks on Virtualized Systems, Hardening the Operating System, Preventing Attacks that Target the Web Browser, Hardening Web Servers, Protecting Systems from Communications-Based Attacks, Applying Software Security Applications.

UNIT-II:

Network Vulnerabilities and Attacks: Network Vulnerabilities, Categories of Attacks, Methods of Network Attacks.

Network Defenses: Crafting a Secure Network, Applying Network Security Devices, Host and Network Intrusion Prevention Systems (HIPS/NIPS), Protocol Analyzers, Internet Content Filters, Integrated Network Security Hardware.

UNIT-III:

Access Control: Access Control Models and Practices, Logical Access Control Methods, Physical Access Control.

Authentication: Definition of Authentication, Authentication Credentials, Extended Authentication Protocols, Remote Authentication and Security.

UNIT-IV:

Vulnerability Assessment: Risk Management, Assessment, and Mitigation, Identifying Vulnerabilities.

Security Audit: Privilege Auditing, Usage Auditing, Monitoring Methodologies and Tools.

UNIT-V:

Cryptography: Introduction to Cryptography, Cryptographic Algorithms, Using Cryptography on Files and Disks, Digital Certificates, Public Key Infrastructure, Key Management.

Text Book:

1. Security+ Guide to Network Security Fundamentals, Third Edition, Mark Ciampa, Cengage Learning.

Reference Books :

1. Principles of Information Security, Michael E. Whitman and Herbert J. Mattord, Cengage Learning.
2. Information Security: The Complete Reference, Rhodes-Ousley, Mark, Second Edition, McGraw-Hill.
3. Information Security: Principles and Practices, Mark S. Merkow, Jim Breithaupt, 2nd Edition, Pearson Education.

	Honor Course	L	T	P	C
		3	0	2	4
SECURE CODING					

Course Objectives:

1. Understanding of the various security attacks and knowledge to recognize and remove common coding errors that lead to vulnerabilities.
2. Knowledge of outline of the techniques for developing a secure application.
3. Recognize opportunities to apply secure coding principles.

Course Outcomes:

At the end of the course, student will be able to

1. List of secure systems and various security attacks
2. Demonstrate the development of process of software leads to secure coding practices
3. Apply Secure programs and various risk in the software's
4. Classify various errors that lead to vulnerabilities
5. Design Real time software and vulnerabilities

UNIT–I:

Introduction-Need for secure systems, Proactive security development process, Security principles to live by and threat modeling.

UNIT–II:

Secure Coding in C-Character strings- String manipulation errors, String Vulnerabilities and exploits Mitigation strategies for strings, Pointers, Mitigation strategies in pointer based vulnerabilities Buffer Overflow based vulnerabilities

UNIT–III:

Secure Coding in C++ and Java-Dynamic memory management, Common errors in dynamic memory management, Memory managers, Double –free vulnerabilities, Integer security, Mitigation strategies

UNIT–IV:

Database and Web Specific Input Issues-Quoting the Input, use of stored procedures, Building SQL statements securely, XSS related attacks and remedies

UNIT–V:

Software Security Engineering-Requirements engineering for secure software: Misuse and abuse cases, SQUARE process model Software security practices and knowledge for architecture and design

Text Book:

1. Michael Howard, David LeBlanc, "Writing Secure Code", Microsoft Press, 2nd Edition, 2003.

References:

1. Robert C. Seacord, "Secure Coding in C and C++", Pearson Education, 2nd edition, 2013.
2. Julia H. Allen, Sean J. Barnum, Robert J. Ellison, Gary McGraw, Nancy R. Mead, "Software Security Engineering: A guide for Project Managers", Addison-Wesley Professional, 2008.

	Honor Course	L	T	P	C
		3	1	0	4
VULNERABILITY ASSESSMENT & PENETRATION TESTING					

Course Objectives:

1. To identify security vulnerabilities and weaknesses in the target applications.
2. To identify how security controls can be improved to prevent hackers gaining access to operating systems and networked environments.
3. To test and exploit systems using various tools.

Course Outcomes:

By the end of this course, the student will be able to

1. Explain Penetration testing phases
2. Illustrate information gathering methodologies
3. Apply System Hacking Techniques in real time applications
4. Explore advanced System hacking
5. Describe Bypassing WLAN Authentication

UNIT-I:

Introduction-Penetration Testing phases/Testing Process, types and Techniques, Blue/Red Teaming, Strategies of Testing, Non-Disclosure Agreement Checklist, Phases of hacking, Open-source/proprietary Pentest Methodologies

UNIT-II :

Information Gathering and Scanning- Information gathering methodologies- Foot printing, Competitive Intelligence- DNS Enumerations- Social Engineering attacks, Port Scanning-Network Scanning- Vulnerability Scanning- NMAP scanning tool- OS Fingerprinting-Enumeration.

UNIT-III :

System Hacking - Password cracking techniques- Key loggers- Escalating privileges- Hiding Files, Double Encoding, Steganography technologies and its Countermeasures. Active and passive sniffing- ARP Poisoning, MAC Flooding- SQL Injection - Error- based, Union-based, Time-based, Blind SQL, Out-of-band. Injection Prevention Techniques.

UNIT- IV :

Advanced System Hacking- Broken Authentication, Sensitive Data Exposure, XML External Entities, Broken Access Code, XSS - Stored, Reflected, DOM Based

UNIT-V :

Wireless Pen test: Wi-Fi Authentication Modes, Bypassing WLAN Authentication, Types of Wireless Encryption, WLAN Encryption Flaws, AP Attack, Attacks on the WLAN Infrastructure, DoS-Layer1, Layer2, Layer 3, DDoS Attack, Client Misassociation, Wireless

Hacking Methodology, Wireless Traffic Analysis

Text Books:

1. Kali Linux 2: Windows Penetration Testing, By Wolf Halton, Bo Weaver , June 2016 Packt Publishing

References:

1. Mastering Modern Web Penetration Testing By Prakhar Prasad,October 2016 Packt Publishing.
2. SQL Injection Attacks and Defense 1st Edition, by Justin Clarke-Salt, Syngress Publication

	Honor Course	L	T	P	C
		3	0	2	4
MALWARE ANALYSIS					

Course Objectives:

1. To understand the purpose of computer infection program.
2. To implement the covert channel and mechanisms.
3. To test and exploit various malware in open-source environment.

Course Outcomes:

At the end of the course, student will be able to

1. Explain the characteristics of Malware and its effects on Computing systems.
2. Predict the given system scenario using the appropriate tools to Identify the vulnerabilities and to perform Malware analysis.
3. Analyze the given Portable Executable and Non-Portable Executable files using Static and dynamic analysis techniques.
4. Demonstrate the Malware functionalities.
5. How to apply anti-reverse engineering in different Applications

UNIT–I:

Malware Basics- General Aspect of Computer infection program, Non Self Reproducing Malware, How does Virus Operate, Virus Nomenclature, Worm Nomenclature, Recent Malware Case Studies.

UNIT– II:

Basic Analysis- Antivirus Scanning, x86 Disassembly, Hashing, Finding Strings, Packed Malware, PE File Format, Linked Libraries & Functions, PE Header File &Section.

UNIT–III:

Advanced Static & Dynamic Analysis-IDA Pro, Recognizing C code constructs, Analyzing malicious windows program, Debugging, OllyDbg, Kernel Debugging with WinDbg, Malware Focused Network Signatures.

UNIT–IV:

Malware Functionalities-Malware Behavior, Covert Malware Launch, Data Encoding, Shell code Analysis.

UNIT–V:

Reverse Engineering Malware (REM): REM Methodology, Resources for Reverse-Engineering Malware (REM) Understanding Malware Threats, Malware indicators, Malware Classification, Examining Clam AV-Signatures.

Text Books:

1. Michael Sikorski, Andrew Honig "Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software" publisher Williampollock

References:

1. ErciFiliol, "Computer Viruses: from theory to applications", Springer, 1st edition, 2005.

	Honor Course	L	T	P	C
		3	1	0	4
STATISTICAL MACHINE LEARNING					

Course Objectives:

1. To provide foundational knowledge of statistical and probabilistic methods relevant to machine learning.
2. To familiarize students with both probabilistic and non-probabilistic models for learning from data.
3. To develop the ability to evaluate, interpret, and validate machine learning models using statistical metrics.
4. To equip students with the knowledge of advanced and research-relevant machine learning techniques.
5. To introduce key topics in modern machine learning including Bayesian reasoning, ensemble learning, deep generative models, and reinforcement learning.

Course Outcomes:

1. Apply probability and inference techniques in machine learning tasks.
2. Implement and evaluate regression/classification models statistically.
3. Perform Bayesian reasoning and probabilistic model updates.
4. Use kernel and ensemble methods for complex datasets.
5. Explore advanced topics like Bayesian networks, deep generative modeling, and reinforcement learning.

UNIT I:

Introduction and Foundations- What is Machine Learning? ,Types of Learning: Supervised, Unsupervised, Reinforcement, Basics of Statistical Inference, Probability Theory Fundamentals, Random Variables, Expectation & Variance, Bayes' Theorem, Maximum Likelihood Estimation (MLE), Bias-Variance Tradeoff

UNIT II:

Linear Models for Regression and Classification-Linear Regression, Least Squares Estimation, Regularization Techniques (Ridge & Lasso), Logistic Regression, Generalized Linear Models (GLMs), Probabilistic Interpretations, Gradient Descent Optimization, Evaluation Metrics: RMSE, MAE, Accuracy, Precision, Recall, ROC Curve, AUC

UNIT III:

Bayesian Learning and Inference- Bayesian Decision Theory, Prior–Likelihood–Posterior Fundamentals, Maximum A Posteriori (MAP), Naïve Bayes Classification, Gaussian Distribution, Bayesian Linear Regression, Conjugate Priors, Model Evidence, Bayesian Occam's Razor

UNIT IV:

Non-Linear Models and Kernel Methods- Decision Trees, Random Forests, k-Nearest Neighbors, Support Vector Machines, Kernel Trick, Polynomial & RBF Kernels, Bias-Variance in Non-linear Models, Cross-Validation Techniques, Hyperparameter Tuning

UNIT V:

Unsupervised Learning & Dimensionality Reduction- k-Means Clustering, Gaussian Mixture Models (GMM), Expectation-Maximization (EM), Hierarchical Clustering, Principal Component Analysis (PCA), t-SNE, Anomaly Detection Techniques, Model Selection Criteria (AIC, BIC), Intro to Graphical Models

UNIT VI:

Advanced Topics- Markov Chain Monte Carlo (MCMC), Gibbs Sampling, Variational Inference, Ensemble Learning (Bagging, Boosting, Stacking), Gradient Boosting (XGBoost, LightGBM), Gaussian Processes, Bayesian Neural Networks, Deep Generative Models (VAEs, GANs), Information Theory Essentials (Entropy, KL-Divergence, Mutual Info), Semi-Supervised Learning, Active Learning, Multi-Task & Transfer Learning, Meta-Learning, Fairness in ML, Explainable AI (XAI), Causal Inference, Reinforcement Learning (Q-Learning, Policy Gradients)

Textbooks:

1. **Kevin P. Murphy** – *Probabilistic Machine Learning: An Introduction*, **MIT Press**, Volume 1 (Mar 2022)
2. **Christopher M. Bishop** – *Pattern Recognition and Machine Learning*, updated paperback, **BookRivers** (India, Apr 2025)

References:

1. **Kevin P. Murphy** – *Probabilistic Machine Learning: Advanced Topics*, **MIT Press**, Volume 2 (2022)
2. **Tom M. Mitchell** – *Machine Learning*, **McGraw-Hill Education**, foundational text (1997; widely used)
3. **T. Hastie, R. Tibshirani & J. Friedman** – *The Elements of Statistical Learning*, **Springer**, 2nd Ed. (2009)
4. **Shai Shalev-Shwartz & Shai Ben-David** – *Understanding Machine Learning: From Theory to Algorithms*, **Cambridge Univ. Press**, 2014
5. **David Barber** – *Bayesian Reasoning and Machine Learning*, **Cambridge Univ. Press**, 2012

	Honor Course	L	T	P	C
		3	1	0	4
REINFORCEMENT LEARNING					

Course Objectives

1. To introduce the fundamental concepts and mathematical foundations of Reinforcement Learning (RL).
2. To explain the RL framework including agents, environments, states, actions, and rewards.
3. To provide hands-on understanding of classical RL algorithms such as value iteration, Q-learning, and SARSA.
4. To explore the trade-off between exploration and exploitation in decision-making under uncertainty.
5. To introduce function approximation techniques and their integration with deep learning in RL.
6. To study advanced RL techniques including policy gradients, actor-critic methods, and deep reinforcement learning.
7. To expose learners to hierarchical RL, multi-agent settings, and real-world RL applications.
8. To highlight current trends, challenges, evaluation methods, and ethical considerations in the deployment of RL systems.

Course Outcomes:

1. Formulate real-world problems as RL setups (states, actions, rewards).
2. Implement and analyze classical RL algorithms (value iteration, Q-learning, SARSA).
3. Apply exploration–exploitation strategies in bandit and MDP settings.
4. Utilize function approximation and deep neural networks in RL.
5. Explore policy gradient methods and modern deep RL algorithms.

UNIT I:

Foundations of Reinforcement Learning-Introduction to Reinforcement Learning, Elements of RL framework, Multi-armed bandits: k-armed bandit problems, actions-value methods, exploration vs exploitation (UCB, gradient bandit), Markov Decision Processes (MDP), Bellman equations, Dynamic Programming: policy evaluation, policy iteration, value iteration

UNIT II:

Model-Free Learning and Temporal-Difference Methods-Monte Carlo methods (first-visit, every-visit, control), Temporal-Difference Learning (TD(0)), SARSA, Q-Learning, Double Q-Learning, n-step TD methods, eligibility traces (TD(λ)), off-policy vs on-policy learning

UNIT III:

Function Approximation and Deep RL-Linear value-function approximation, semi-gradient TD methods, neural network architectures for RL, Deep Q-Networks (DQN), experience replay, target networks, deep extensions (Double DQN, Dueling DQN), Policy gradient methods: REINFORCE, actor-critic methods, Advantage Actor-Critic (A2C), Deep Deterministic Policy Gradient (DDPG), PPO

UNIT IV:

Hierarchical & Advanced Reinforcement Learning- Extensions: Semi-Markov Decision Processes (semi-MDPs), options framework, hierarchical reinforcement learning, POMDPs, multi-agent RL settings (cooperative and competitive), contextual bandits, trust region methods, recent developments

UNIT V:

Case Studies, Ethics, and Applications-Applications of RL: games (Atari, Go/AlphaGo), robotics, recommendation systems; RL from human feedback (RLHF); challenges like sample complexity, stability, fairness; evaluation metrics: regret, convergence, computational cost; ethical implications and responsible deployment

Textbooks :

1. **Richard S. Sutton & Andrew G. Barto** – *Reinforcement Learning: An Introduction*, 2nd Edition, **MIT Press**, 2018
2. **Dimitri P. Bertsekas** – *Reinforcement Learning and Optimal Control*, 1st Edition, **Athena Scientific**, 2019

References :

1. **Dimitri P. Bertsekas** – *Dynamic Programming and Optimal Control*, Volumes I & II, **Athena Scientific**, 4th Edition, 2017
2. **Csaba Szepesvári** – *Algorithms of Reinforcement Learning*, **Morgan & Claypool**, 2010
3. **Warren B. Powell** – *Reinforcement Learning and Stochastic Optimization*, **Wiley**, 2022
4. **Aske Plaat** – *Deep Reinforcement Learning: A Textbook* (2022 draft), covers advanced topics like hierarchical RL and multi-agent RL

	Honor Course	L	T	P	C
		3	1	0	4
PROGRAMMING FOUNDATIONS WITH GENAI TOOLS					

Course Objectives:

By the end of the course, students will be able to:

1. Understand the basics of programming and the capabilities of modern Generative AI tools in aiding software development.
2. Learn data representation, manipulation, and control structures in Python.
3. Develop skills in code comprehension, testing, and debugging with both human and AI assistance.
4. Compare programming paradigms between Python and C, and understand low-level implementation concepts.
5. Explore performance improvements and gain a preview of advanced programming concepts such as data structures and algorithms.

Course Outcomes:

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

1. Explain the fundamentals of programming and evaluate the role of Generative AI tools in enhancing programming productivity.
2. Apply Python constructs to represent and manipulate data, and design modular code using functions, conditionals, and recursion.
3. Analyze and debug Python programs effectively, including AI-generated code, using appropriate testing and troubleshooting techniques.
4. Compare Python with C, implement basic programs in both languages, and understand core concepts such as memory management and data structures.
5. Identify and optimize performance-critical code segments, and gain foundational insights into data structures and algorithms.

UNIT–I:

Introduction to Programming and GenAI Tools

What is programming?, Role and effectiveness of modern Generative AI tools in programming, Optional: Installing and configuring GenAI tools on your system

UNIT–II:

Data Representation and Manipulation in Python

Fundamental data types and problem-solving mismatches, Objects, expressions, and assignments, Error handling in Python, Using functions to solve sub-problems, Principles of good function design, Introduction to conditionals and recursion

UNIT–III:

Code Comprehension and Debugging in Python

Critiquing code (manual and AI-generated), Testing code and identifying potential errors, Lists and iteration, Asking clarifying questions in problem statements, Debugging using manual and AI-assisted techniques, Introduction to dictionaries

UNIT–IV:

Learning and Comparing Programming Languages (Python vs C)

Key differences between Python and C, Introduction to arrays and structs in C, Memory management basics in C, Implementing a simplified version of Python lists in C

UNIT–V:

Performance-Oriented Programming and Algorithms

Identifying performance bottlenecks in Python code, Translating performance-critical code from Python to C, Introduction to Data Structures and Algorithms (preview)

Textbooks :

1. **Paul Deitel, Harvey Deitel**, *Python for Programmers*, **Pearson Education, 2nd Edition, 2022, ISBN: 978-0136905662**
2. **Brian W. Kernighan, Dennis M. Ritchie**, *The C Programming Language*, **PHI Learning, 2nd Edition (Indian Reprint, Latest 2021), ISBN: 978-8177581530**
3. **Luciano Ramalho**, *Fluent Python: Clear, Concise, and Effective Programming*, **O'Reilly Media, 2nd Edition, 2022, ISBN: 978-1492056355**
4. **Mark Lutz**, *Learning Python*, **O'Reilly Media, 5th Edition, Latest Reprint 2023, ISBN: 978-9351102014**
5. **Sinan Ozdemir**, *Rust and Python Programming with Generative AI Integration*, **Packt Publishing, 1st Edition, 2024, ISBN: 978-1805124796**

References :

1. Python 3 documentation (https://docs.python.org/3/)
2. The GNU C Reference Manual (https://www.gnu.org/ software/gnu-c-manual/gnu-c-manual.html)
3. Learn AI-Assisted Python Programming (Leo Porter and Daniel Zingaro, Manning Sept 2023, ISBN 9781633437784

	Honor Course	L	T	P	C
		3	0	2	4
ADVANCED COMPUTER VISION					

Course Objectives:

1. Able to apply the core theories and algorithms of computer vision and video processing
2. Understand the state-of-the-art of computer vision and image/video processing,
3. Apply the applications such as vision-based modeling and interaction.

Course Outcomes:

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

1. Identify basic concepts, terminology, theories, models and methods in the field of computer vision.
2. Able to know principles of human visual system.
3. Understanding the advanced methods of computer vision related to GAN, RNN, Deep Dream implementation, LeNet and MNIST etc...
4. Apply a design of a computer vision system for a specific problem.
5. Apply applications of RNN in real time applications.

UNIT – I:

Introduction to Deep Learning, Tensor flow and Keras: What is Deep learning? Why Deep learning, Advantages, and limitations of Deep learning. Tensor flow basics, how to build Deep learning models with Keras and Tensor flow as back end. Tensor board for visualizations.

UNIT - II:

CNN for Vision Tasks: Introduction to CNN, Deep Convolutional networks, LeNet, VGG16Net, and Classification of MNIST hand written digits by CNN and FCNN models.

UNIT - III:

Generative Adversal Networks(GAN's): What is GAN?, DGAN, Some interesting GAN structures, SRGAN, Cycle GAN, info GAN.MNIST using GAN in Tensor flow.

UNIT - IV:

Recurrent Neural Networks: The basic RNN, RNN Cell, RNN variants, RNN topologes, Example applications of RNN. Image captioning and Annotation.

UNIT - V:

Deep Dream and Neural Style Transfer: How the Deep dream algorithm works, Deepdeam implementation in keras and tensor flow. Neural Style Transfer: Content loss, Style loss, Total varianlosses, network training.

Text Books:

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systemsby AurélienGéron,Orielly.
2. Deep Learning with Python 1st Edition by François Chollet,MannigPublicatons.
3. Mastering Computer Vision with TensorFlow 2.x: Build advanced computer vision applications using machine learning and deep learning techniquesby Krishnendu Kar ,Packt Publications.
4. Deep Learning with TensorFlow 2 and Keras: Regression, ConvNets, GANs, RNNs,NLP, and more with TensorFlow 2 and the Keras API, 2nd Edition

References:

1. Richard Szeliksy "Computer Vision: Algorithms and Applications" (http://szeliski.org/Book/)
2. Haralick& Shapiro, "Computer and Robot Vision", Vol II
3. G_erardMedioni and Sing Bing Kang "Emerging topics in computer vision"
4. Emanuele Trucco and AllessandroVerri "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998.
5. Olivier Faugeras, "Three-Dimensional Computer Vision", The MIT Press, 1993

	Honor Course	L	T	P	C
		3	1	0	4
MATHEMATICAL ESSENTIAL FOR DATA SCIENCE					

Course Objectives:

1. Recall the basics of sets, natural numbers, integers, rational numbers, and real numbers.
2. Learn to use the coordinate system, and plot straight lines.
3. Identify the properties and differences between linear, quadratic, polynomial, exponential, and logarithmic functions.

Course Outcomes:

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

1. Demonstrate understanding of basic mathematical concepts in data science, relating to linear algebra, probability, and calculus.
2. Employ methods related to these concepts in a variety of data science applications.
3. Apply logical thinking to problem-solving in context.
4. Use appropriate technology to aid problem-solving and data analysis.
5. Demonstrate skills in writing mathematics.

UNIT – I:

Set Theory - Number system, Sets and their operations;

Relations and functions - Relations and their types, Functions and their types, Rectangular coordinate system

UNIT – II:

Straight Lines- Slope of a line, Parallel and perpendicular lines, Representations of a Line, General equations of a line, Straight-line fit

Quadratic Functions - Quadratic functions, Minima, maxima, vertex, and slope, Quadratic Equations

UNIT – III:

Algebra of Polynomials - Addition, subtraction, multiplication, and division, Algorithms

UNIT – IV:

Graphs of Polynomials - X-intercepts, multiplicities, end behavior, and turning points, Graphing & polynomial creation

Functions - Horizontal and vertical line tests, Exponential functions, Composite functions, Inverse functions

Logarithmic Functions - Properties, Graphs, Exponential equations, Logarithmic equations

UNIT – V:

Graph Theory - Representation of graphs, Breadth-first search, Depth-first search, Applications of BFS and DFS

Directed Acyclic Graphs - Complexity of BFS and DFS, Topological sorting and longest path, Transitive closure, Matrix multiplication

Graph theory Algorithms - Single source shortest paths, Dijkstra's algorithm, Bellman-Ford algorithm, All-pairs shortest paths, Floyd–Warshall algorithm, Minimum cost spanning trees, Prim's algorithm, Kruskal's algorithm

Text Book:

1. Introductory Algebra: a real-world approach (4th Edition) - by Ignacio Bello

References:

1. Mathematical Foundations Of Data Science Using Rby Emmert-Streib Frank.

	Honor Course	L	T	P	C
		3	1	0	4
INTRODUCTION TO DATA SCIENCE					

Course Objectives:

1. The course teaches critical concepts and skills in computer programming and statistical inference, in conjunction with hands-on analysis of real-world datasets, including economic data, document collections, geographical data, and social networks.
2. It delves into social issues surrounding data analysis such as privacy and design.
3. Explain out-of-sample evaluation metrics

Course Outcomes:

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

1. Apply dimensionality reduction tools such as principle component analysis
2. Evaluate outcomes and make decisions based on data
3. Understand how to Use exploratory tools such as clustering and visualization tools to analyze data.
4. Apply dimensionality reduction tools such as principle component analysis
5. Able to know how to perform basic analysis of network data.

UNIT – I:

Introduction

Introduction to Data Science – Evolution of Data Science – Data Science Roles – Stages in a Data Science Project – Applications of Data Science in various fields – Data Security Issues.

UNIT – II:

Data Collection and Data Pre-Processing

Data Collection Strategies – Data Pre-Processing Overview – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization.

UNIT – III:

Exploratory Data Analytics

Descriptive Statistics – Mean Standard Deviation, Skewness and Kurtosis – Box Plots – Pivot Table – Heat Map Correlation Statistics – ANOVA.

UNIT – IV:

Model Development

Simple and Multiple Regression – Model Evaluation using Visualization – Residual Plot – Distribution Plot – Polynomial Regression and Pipelines – Measures for In-sample Evaluation – Prediction and Decision Making.

UNIT – V:

Model Evaluation

Generalization Error – Out-of-Sample Evaluation Metrics – Cross Validation – Over fitting –Under Fitting and Model Selection – Prediction by using Ridge Regression – Testing Multiple Parameters by using Grid Search.

Text Books:

1. Data Science for Beginners, by Andrew Park
2. The Art of Data Science — A Guide for Anyone Who Works With Data, by Roger D. Peng and Elizabeth Matsui.

References:

1. JojoMoolayil, “Smarter Decisions : The Intersection of IoT and Data Science”,PACKT, 2016.
2. Cathy O’Neil and Rachel Schutt , “Doing Data Science”, O'Reilly, 2015.
3. David Dietrich, Barry Heller, Beibei Yang, “Data Science and Big data Analytics”,EMC 2013
4. Raj, Pethuru, “Handbook of Research on Cloud Infrastructures for Big DataAnalytics”, IGI Global.

	Honor Course	**L**	**T**	**P**	**C**

		3	1	0	4
DATA ANALYTICS AND VISUALIZATION					

Course Objectives:

1. To demonstrate expert knowledge of data analysis, statistics, tools, techniques and technologies of data analytics and Visualization.
2. To enable learners to develop knowledge and skills in current and emerging areas of data analytics and Visualization.
3. To formulate and implement a novel research idea and conduct research in the field of data analytics and Visualization.

Course Outcomes:

After completing the course, student will be able to:

1. Present data with visual representations for your target audience, task, and data;
2. Identify appropriate data visualization techniques given particular requirements imposed by the data;
3. Display types, Geospatial displays, Interactivity
4. Data Definitions and Analysis Techniques
5. Implement the analytic algorithms and Basic analysis techniques

UNIT -1:

Introduction and Tableau Primer: Introduction to data visualization Data for data graphics Tableau introduction

UNIT-2:

Design Principles: Design principles Categorical, time series, and statistical data graphics

UNIT-3:

Display types, Geo spatial displays, Interactivity: Storytelling Multivariate displays, Geospatial displays, Dashboards, interactive and animated displays

UNIT-4:

Data Definitions and Analysis Techniques: Elements, Variables, and Data categorization, Levels of Measurement, Data management and indexing, Introduction to statistical learning.

Descriptive Statistics: Measures of central tendency, Measures of location of dispersions.

UNIT-5:

Basic analysis techniques: Statistical hypothesis generation and testing, Chi-Square test,t-Test,Analysis of variance, Correlation analysis, Maximum likelihood test.

Text Books:

1. Sosulski, K. (2018). Data Visualization Made Simple: Insights into BecomingVisual. New York: Routledge.

2. Probability & Statistics for Engineers & Scientists (9th Edn.), Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers and Keying Ye, Prentice Hall Inc.
3. The Elements of Statistical Learning, Data Mining, Inference, and Prediction (2nd Edn.), Trevor Hastie Robert Tibshirani Jerome Friedman, Springer, 2014

References:

1. An Introduction to Statistical Learning: with Applications in R, G James, D. Witten, T Hastie, and R. Tibshirani, Springer, 2013
2. Software for Data Analysis: Programming with R (Statistics and Computing), John M. Chambers, Springer
3. Mining Massive Data Sets, A. Rajaraman and J. Ullman, Cambridge University Press, 2012
4. Advances in Complex Data Modeling and Computational Methods in Statistics, Anna Maria Paganoni and Piercesare Secchi, Springer, 2013

Optional readings:

1. Few, S. (2012). Show me the numbers: Designing tables and graphs toenlighten. Burlingame, CA: Analytics Press.
2. Few, S. (2006). Information dashboard design: The effective visualcommunication of data. Sebastopol: O'Reilly.
3. Ware, C & Kaufman, M. (2008). Visual thinking for design. Burlington: MorganKaufmann Publishers.
4. Wong, D. (2011). The Wall Street Journal guide to information graphics: The dosand don'ts of presenting data, facts and figures. New York: W.W. Norton& Company.
5. Yau, N. (2011). Visualize This: The FlowingData Guide to Design, Visualization,and Statistics. Indianapolis: O'Reilly.
6. Yau, N. (2013). Data Points: Visualization that means something. Indianapolis: O'Reilly.

	Honor Course	L	T	P	C
		3	1	0	4
GRAPH ANALYTICS					

Course Objectives:

1. To provide a solid foundation in graph theory concepts and graph representations.
2. To familiarize students with classical graph algorithms and their applications in real-world problems.
3. To introduce advanced topics such as community detection, centrality measures, and network analysis techniques.
4. To explore graph-based machine learning models, including node embeddings and Graph Neural Networks (GNNs).
5. To equip students with hands-on skills using modern graph analytics tools and frameworks like Python, Neo4j, and TigerGraph.
6. To enable analysis of large-scale and complex graph data for applications in diverse domains such as social networks, fraud detection, and knowledge graphs**.**

Course Outcomes:

1. Demonstrate knowledge of graph representations and core graph properties.
2. Implement classical graph algorithms (traversals, shortest paths, centrality).
3. Apply community detection and clustering methods to real-world graphs.
4. Use node embeddings and graph neural networks for analytics tasks.
5. Analyze graphs using Python/Neo4j frameworks and handle large-scale graph data.

UNIT I:

Introduction to Graph Theory & Representations- Types of graphs (directed, undirected, weighted), Graph representations (adjacency matrix, list, edge list), Graph properties & terminology, Graph operations and transformations

UNIT II:

Graph Algorithms & Centrality Measures- Breadth-First Search (BFS), Depth-First Search (DFS), Shortest path algorithms (Dijkstra, Bellman-Ford), Centrality measures (Degree, Betweenness, Closeness, Eigenvector centrality, PageRank), Applications to social networks and recommendation systems

UNIT III:

Community Detection & Network Analysis - Community detection methods (Louvain, Girvan-Newman), Modularity, Network motifs, Structural balance theory, Triadic closure, Subgraph analysis, Case-study applications on social and biological networks

UNIT IV:

Graph-based Machine Learning & Embeddings- Graph-based ML: Graph Convolutional Networks (GCNs), Graph Neural Networks (GNNs), Node embedding methods (node2vec, DeepWalk), Graph kernels, Link prediction, Node classification

UNIT V:

Large-Scale & Parallel Graph Analytics - Random graph models: Erdos–Rényi, Small-world, Scale-free; Spectral graph analysis, GraphBLAS and linear-algebraic graph computation, Parallel graph analytics frameworks (GraphX/Pregel), scalability considerations

UNIT VI:

Applications, Tools & Case Studies- Graph DBs (Neo4j, Cypher queries), Graph analytics pipelines, Applications: fraud detection, social media mining, citation networks, knowledge graphs; Integration into data science workflows using Python, Neo4j, TigerGraph

Text books :

1. **Timothy Eastridge** – *Graph Data Science with Python and Neo4j*, **Enterprise Strategies**, 2024
2. **Tomaž Bratanic** – *Graph Algorithms for Data Science* (with Neo4j examples), **2024** (self-published)

Reference Books :

1. **Victor Lee, Phuc Nguyen & Alexander Thomas** – *Graph-Powered Analytics and Machine Learning with TigerGraph*, **O'Reilly Media**, 2023
2. **Bonia diestel et al.** – *Graph Theory and Applications*, **Springer**, 2008 (classic with strong fundamentals)
3. **Albert-László Barabási** – *Network Science*, **Cambridge Univ. Press**, 2016
4. **M. E. J. Newman** – *Networks: An Introduction*, **Oxford University Press**, 2010

JNTUGV- COLLEGE OF ENGINEERING VIZIANAGARAM

DEPARTMENT OF INFORMATION TECHNOLOGY

B.TECH- IT (R23)-MINORS

MINOR DEGREE IN INFORMATION TECHNOLOGY

(For Non-CSE / IT Students)

(I) Artificial Intelligence

S.No	Subject Title	L	T	P	C
1	Introduction to Artificial Intelligence	3	0	0	3
2	Mathematics for Machine Learning	3	0	0	3
3	Machine Learning	3	0	0	3
4	Deep Learning	3	0	0	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/ SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(II) Computer Security

S.No	Subject Title	L	T	P	C
1	Cyber Security	3	0	0	3
2	Cyber Crime Investigation and Digital Forensics	3	0	0	3
3	Cryptography and Applications	3	0	0	3
4	Blockchain Technology	3	0	0	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(III) Programming and Web Development

S.No	Subject Title	L	T	P	C
1	Object Oriented Programming through Java	2	0	2	3
2	Python Programming	2	0	2	3
3	Basic Web Designing	2	0	2	3
4	Advanced Web Technologies	2	0	2	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

(IV) Advanced Computing

S.No	Subject Title	L	T	P	C
1	Computer Organization and Architecture	3	0	0	3
2	Distributed Systems	3	0	0	3
3	Cloud Computing	3	0	0	3
4	Quantum Computing	3	0	0	3
5	03 MOOCS courses @ 2credits each (8-week course) 02 MOOCS courses @ 3credits each (12-week course) (Any CSE/IT related **Program Core subject** from NPTEL/SWAYAM course other than the courses listed above needs to be taken)				6
				Total	**18**

Note: Students who have registered for minors program can opt a maximum of two (02) courses per semester.

	Minor Course	L	T	P	C
		3	1	0	4
INTRODUCTION TO ARTIFICIAL INTELLIGENCE					

Course Objectives:

1. To have a basic proficiency in a traditional AI language including an ability to write simple to intermediate programs and an ability to understand code written in that language
2. To understand the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other topics such as minimax, resolution, etc. that play an important role in AI programs
3. To have a basic understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and robotics, expert systems, and planning.

Course Outcomes:

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

1. Outline problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem
2. Apply the language/framework of different AI methods for a given problem
3. Implement basic AI algorithms- standard search algorithms or dynamic programming
4. Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports
5. Design Expert Systems using fuzzy logic theory

UNIT- I:

Introduction: history, intelligent systems, foundations of AI, applications, tic-tac-toe game playing, development of AI languages, current trends.

UNIT -II:

Problem Solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative deepening A*, constraint satisfaction.

Problem reduction and game playing: Introduction, problem reduction, game playing, alpha beta pruning, two-player perfect information games.

UNIT –III:

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic.

UNIT -IV:

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames.

Advanced knowledge representation techniques: Introduction, conceptual dependency theory, script structure, CYC theory, case grammars, semantic web

UNIT–V:

Expert system and applications: Introduction phases in building expert systems, expert system versus traditional systems Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory .

Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

Text Books:

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning
2. Artificial intelligence, A modern Approach , 2nded, Stuart Russel, Peter Norvig,PEA

References:

1. Artificial Intelligence- Deepak Khemani, TMH,2013
2. Introduction to Artificial Intelligence, Patterson,PHI
3. Atificial intelligence, structures and Strategies for Complex problem solving,-George F Lugar, 5thed, PEA

E-Resources:

1. https://nptel.ac.in/courses/106/105/106105077/
2. http://aima.cs.berkeley.edu/

	Minor Course	L	T	P	C
		3	1	0	4
MATHEMATICS FOR MACHINE LEARNING					

Course Objectives:

1. The purpose of this course is to provide a mathematically rigorous introduction to these developments with emphasis on methods and their analysis.
2. Explain and apply matrix decomposition techniques
3. Explain parameter estimation using the Maximum Likelihood method

Course Outcomes:

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

1. Understand the strengths and weaknesses of many popular machine learning approaches.
2. Justify the underlying mathematical relationships within and across Machine Learning algorithms.
3. Evaluate the several areas of mathematics beyond calculus
4. Solve problems in a range of mathematical applications
5. Apply various methods to compute the probabilities of events, Analyze and interpret statistical data using appropriate probability distributions.

UNIT-I:

Linear Algebra: Systems of Linear Equations, Matrices, Solving systems of linear equations, Vector Spaces, Linear Independence, Basis and Rank, Linear Mappings.

Analytic Geometry: Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Orthonormal Basis, Orthogonal Complement, Inner Product of Functions, Orthogonal Projections.

UNIT-II:

Matrix Decompositions: Determinant and Trace, Eigen values and Eigen vectors, Cholesky Decomposition, Eigen decomposition and Diagonalization, Singular Value Decomposition, Matrix Approximation.

Vector Calculus: Differentiation of Univariate Functions, Partial differentiation and Gradients, Gradients of vector valued functions, Gradients of Matrices, Useful identities for computing gradients, Back propagation and Automatic Differentiation

UNIT-III:

Probability and Distributions: Construction of a Probability space, Discrete and Continuous probabilities, sum rule, product rule and Bayes Theorem, Summary statistics and Independence, Gaussian Distribution.

Continuous Optimization: Optimization using Gradient Descent, Constrained optimization and Lagrange Multipliers, Convex Optimization.

UNIT-IV:

Linear Regression: Problem Formulation, Parameter Estimation, Bayesian Linear Regression, Maximum Likelihood as Orthogonal Projection.

Dimensionality Reduction with Principal Component Analysis: Problem setting, Maximum Variance Perspective, Projection Perspective, Eigenvector computation and Low Rank Approximations, PCA in High Dimensions, Latent Variable Perspective.

UNIT-V:

Density Estimation with Gaussian Mixture Models: Gaussian Mixture Model, Parameter Learning via Maximum Likelihood, EM Algorithm, Latent-Variable Perspective.

Classification with Support Vector Machines: Separating Hyper planes, Primal Support Vector Machine, Dual Support Vector Machine, Kernels, Numerical Solution.

Text Books:

1. https://mml-book.github.io/book/mml-book.pdf - c 2021 M. P. Deisenroth, A. A. Faisal, C. S. Ong. Published by Cambridge University Press (2020).

References:

1. https://www.youtube.com/watch?v=1VSZtNYMntM

	Minor Course	L	T	P	C
		3	1	0	4
MACHINE LEARNING					

Course Objectives:

1. To learn well -known machine learning algorithms
2. To evaluate and compare the performance of various machine learning algorithms
3. Able to differentiate regression models and distance based models and ANNS.

Course Outcomes:

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

1. Recognize the characteristics of machine learning algorithms and their applications to real world problems
2. Able to differentiate linear and logistic regressions.
3. Able to write and evaluate hypothesis
4. Understand the concepts of Artificial neural networks
5. Can apply kernel methods to solve real world problems.

UNIT I:

Introduction: Well -posed learning problems, designing a learning system, Perspectives, and issues in machine learning. Concept learning and the general to specific ordering – Introduction, A concept learning task, Concept learning as search, Find -S: finding a maximally specific hypothesis, Version spaces and the candidate elimination algorithm, Remarks on version spaces and candidate elimination, Inductive bias.

UNIT II:

Linear Regression & Logistic Regression- Predicting Numeric values : Regression - Finding the best fit lines with linear regression, locally weighted linear regression, Shrinking Coefficients, The bias / Variance tradeoff.

Logistic Regression: Classification with logistic regression and the sigmoid function, Using optimization to find the best regression coefficients.

UNIT III:

Artificial Neural Networks : Introduction, Neural network representation, Appropriate problems for neural network learning, Perceptions, Multilayer networks and the back propagation algorithm, Remarks on the back propagation algorithm, An illustrative example face recognition, Advanced topics in artificial neural networks

UNIT IV:

Evaluation Hypotheses: Motivation, Estimation hypothesis accuracy, Basics of sampling theory, A general approach for deriving confidence intervals, Difference in error of two hypotheses, Comparing learning algorithms.

UNIT V:

Support Vector Machines: Separating data with the maximum margin, finding the maximum margin, efficient optimization with SMO algorithm, speeding up optimization with full platt SMO, Using Kernels for more Complex data.

Text Books:

1. Machine Learning ,Tom M. Mitchell, MGH
2. Machine Learning in Action, Peter Harington, 2012, Cengage.

References:

1. Introduction to Machine Learning, EthemAlpaydin, PHI, 2004
2. A course in Machine Learning , Hall Daum'e III

	Minor Course	L	T	P	C
		3	1	0	4
DEEP LEARNING					

Course Objectives:

At the end of the course, the students will be expected to:

1. Learn deep learning methods for working with sequential data,
2. Learn deep recurrent and memory networks,
3. Learn deep Turing machines

Course Outcomes:

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

1. Demonstrate the basic concepts fundamental learning techniques and layers.
2. Discuss the Neural Network training, various random models.
3. Explain different types of deep learning network models.
4. Classify the Probabilistic Neural Networks and Sequence model neural networks.
5. Implement tools on Deep Learning techniques.

UNIT I:

Introduction: Various paradigms of learning problems, Perspectives and Issues in deep learning framework, review of fundamental learning techniques.

Feed forward neural network: Artificial Neural Network, activation function, multi-layer neural network.

UNIT II:

Training Neural Network: Risk minimization, loss function, back propagation, regularization, model selection, and optimization.

Conditional Random Fields: Linear chain, partition function, Markov network, Belief propagation, Training CRFs, Hidden Markov Model, Entropy.

UNIT III:

Deep Learning: Deep Feed Forward network, regularizations, training deep models, dropouts, Convolution Neural Network, Recurrent Neural Network, and Deep Belief Network.

UNIT IV:

Probabilistic Neural Network: Hopfield Net, Boltzmann machine, RBMs, Sigmoid net, Auto encoders.

Sequence Modeling: LSTM, Gated RNNs & Deep Generative Models

UNIT V:

Applications: Object recognition, sparse coding, computer vision, natural language processing.

Introduction to Deep Learning Tools: Caffe, Theano, Torch.

Text Books:

1. Good fellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016..
2. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006.

References:

1. Artificial Neural Networks, Yeg nanarayana, B., PHI Learning Pvt. Ltd, 2009.
2. Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.
3. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education, 2004.

	Minor Course	L	T	P	C
		3	1	0	4
CYBER SECURITY					

Course Objectives:

In this course, the student will learn about

1. The essential building blocks and basic concepts around cyber security such as Confidentiality, Integrity, Availability, Authentication, Authorization, Vulnerability, Threat & Risk and so on.
2. Analyze various browser-based attacks
3. Explore strategic network defense mechanisms

Course Outcomes:

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

1. Illustrate the broad set of technical, social & political aspects of Cyber Security and security management methods to maintain security protection
2. Appreciate the vulnerabilities and threats posed by criminals, terrorist and nation states to national infrastructure
3. Illustrate the nature of secure software development and operating systems
4. Demonstrate the role security management plays in cyber security defense and legal and social issues at play in developing solutions
5. Assess privacy concerns in data mining and web-based applications

UNIT –I:

Introduction: Introduction to Computer Security, Threats, Harm, Vulnerabilities, Controls, Authentication, AccessControl, and Cryptography, Authentication, Access Control, Cryptography.

Programs and Programming: Unintentional (Non-malicious) Programming Oversights, Malicious Code—Malware, Countermeasures.

UNIT –II:

Web Security: User Side, Browser Attacks, Web Attacks Targeting Users, Obtaining User or Website Data, Email Attacks.

Operating Systems Security: Security in Operating Systems, Security in the Design of Operating Systems, Root kit.

UNIT -III:

Network Security: Network Concepts, Threats to Network Communications, Wireless Network Security, Denial of Service, Distributed Denial-of-Service Strategic Defenses: Security Countermeasures, Cryptography in Network Security, Firewalls, Intrusion Detection and Prevention Systems, Network Management .

Cloud Computing and Security: Cloud Computing Concepts, Moving to the Cloud, Cloud

Security Tools and Techniques, Cloud Identity Management, Securing IaaS.

UNIT- IV:

Privacy: Privacy Concepts, Privacy Principles and Policies, Authentication and Privacy, Data Mining, Privacy on the Web, Email Security, Privacy Impacts of Emerging Technologies, Where the Field Is Headed.

Management and Incidents: Security Planning, Business Continuity Planning, Handling Incidents, Risk Analysis, Dealing with Disaster.

UNIT –V:

Legal Issues and Ethics: Protecting Programs and Data, Information and the Law, Rights of Employees and Employers, Redress for Software Failures, Computer Crime, Ethical Issues in Computer Security, Incident Analysis with Ethics Emerging Topics: The Internet of Things, Economics, Computerized Elections, Cyber Warfare.

Text Books:

1. Pfleeger, C.P., Security in Computing, Prentice Hall, 2010, 5th edition.
2. Schneier, Bruce. Applied Cryptography, Second Edition, John Wiley & Sons, 1996

References:

1. Rhodes-Ousley, Mark. Information Security: The Complete Reference, Second Edition, Information Security Management: Concepts and Practice, McGraw-Hill, 2013.
2. Whitman, Michael E. and Herbert J. Mattord. Roadmap to Information Security for IT and Infosec Managers. Boston, MA: Course Technology, 2011.

	Minor Course	L	T	P	C
		3	1	0	4
CYBER CRIME INVESTIGATION AND DIGITAL FORENSICS					

Course Objectives:

1. Able to identify security risks and take preventive steps
2. To understand the forensics fundamentals.
3. To understand the evidence capturing process.

Course Outcomes:

At the end of the course, student will be able to

1. Acquire the definition of computer forensics fundamentals.
2. Describe the types of computer forensics technology
3. Analyze various computer forensics systems.
4. Illustrate the methods for data recovery, evidence collection and data seizure.
5. Summarize duplication and preservation of digital evidence.

UNIT–I:

Introduction: Introduction and Overview of Cyber Crime, Nature and Scope of Cyber Crime, Types of Cyber Crime: Social Engineering, Categories of Cyber Crime, Property Cyber Crime.

UNIT–II:

Cyber Crime Issues: Unauthorized Access to Computers, Computer Intrusions, White collar Crimes, Viruses and Malicious Code, Internet Hacking and Cracking, Virus Attacks, Pornography, Software Piracy, Intellectual Property, Mail Bombs, Exploitation ,Stalking and Obscenity in Internet, Digital laws and legislation, Law Enforcement Roles and Responses.

UNIT–III:

Investigation: Introduction to Cyber Crime Investigation, Investigation Tools, e-Discovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidences, Password Cracking.

UNIT–IV:

Digital Forensics: Introduction to Digital Forensics, Forensic Software and Hardware, Analysis and Advanced Tools, Forensic Technology and Practices, Forensic Ballistics and Photography, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Network Forensics.

UNIT– V:

Laws And Acts: Laws and Ethics, Digital Evidence Controls, Evidence Handling Procedures, Basics of Indian Evidence ACT IPC and CrPC ,Electronic Communication Privacy ACT, Legal Policies.

Text Books:

1. Nelson, B., Phillips, A., & Steuart, C., *Guide to Computer Forensics and Investigations*, Cengage Learning, 2018, 6th Edition.
2. Casey, Eoghan, *Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet*, Academic Press, 2011, 3rd Edition.
3. Brenner, Susan W., *Cybercrime: Criminal Threats from Cyberspace*, Praeger Security International, 2010.

	Minor Course	L	T	P	C
		3	1	0	4
CRYPTOGRAPHY AND APPLICATIONS					

Course Objective:

This course aims at training students to master the:

1. The concepts of classical encryption techniques and concepts of finite fields and number theory
2. Working principles and utilities of various cryptographic algorithms including secret key cryptography, hashes and message digests, and public key algorithms.
3. Design issues and working principles of various authentication protocols and PKI standards.

Course Outcomes:

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

1. Identify information security goals and acquire fundamental knowledge on the concepts of finite fields and number theory
2. Compare and apply different encryption and decryption techniques to solve problems related to confidentiality and authentication
3. Apply the knowledge of cryptographic checksums and evaluate the performance of different message digest algorithms for verifying the integrity of varying message sizes.
4. Apply different digital signature algorithms to achieve authentication and create secure applications
5. Apply network security basics, analyze different attacks on networks and evaluate the performance of security protocols like SSL, IPSec, and PGP

UNIT- I:

Introduction to Security: Security Attacks, Security Services, Security Mechanisms, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, a Model for Network Security

Mathematics of Cryptography: Algebraic Structures (Groups, Rings, Fields and Galois Fields), Divisibility and the Division Algorithm, The Euclidean Algorithm, Modular Arithmetic, Prime Numbers, Fermat's and Euler's Theorems, Testing for Primality, The Chinese Remainder Theorem, Discrete Logarithms

UNIT- II:

Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines, Steganography

Block Ciphers: Traditional Block Cipher Structure, The Data Encryption Standard, The

Strength of DES, Block Cipher Design Principles, Advanced Encryption Standard, AES Structure, AES Transformation Functions, AES Key Expansion, Multiple Encryption and Triple DES, Block Cipher Modes of Operation

UNIT- III:

Public-Key Cryptography: Principles of Public-Key Cryptosystems, The RSA Algorithm, Diffie- Hellman Key Exchange, Elgamal Cryptographic System, Elliptic Curve Cryptography, **Cryptographic Hash Functions**: Applications of Cryptographic Hash Functions, Requirements and Security, Secure Hash Algorithm (SHA), **Message Authentication Codes**: Requirements for Message Authentication Codes, HMAC, CMAC

UNIT- IV:

Digital Signatures: Digital Signatures, Elgamal Digital Signature Scheme, Schnorr Digital Signature Scheme, NIST Digital Signature Algorithm, Elliptic Curve Digital Signature Algorithm

Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public-Key Infrastructure

User Authentication: Remote User-Authentication Principles, Remote User-Authentication Using Symmetric Encryption, Kerberos, Remote User-Authentication Using Asymmetric Encryption:

UNIT -V:

Transport-Level Security: Web Security Considerations, Transport Layer Security, Secure Shell (SSH)

Electronic Mail Security: S/MIME, Pretty Good Privacy

IP Security: IP Security Overview, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange

Text Book:

1. Cryptography and Network Security, William Stallings, 8th Edition, Pearson Education

References:

1. Cryptography, Network Security and Cyber Laws, Bernard L. Menezes, Ravinder Kumar, Cengage Learning.
2. Cryptography and Network Security, Behrouz A Forouzan, DebdeepMukhopadhyaya, 3rd Edition, Mc-GrawHill.
3. Network Security Illustrated, Jason Albanese, Wes Sonnenreich, and McGraw Hill.

		L	T	P	C
	Minor Course	3	1	0	4
BLACK CHAIN TECHNOLOGY					

Course Objectives:

1. To provide conceptual understanding of the function of Block chain as a method of securing distributed ledgers.
2. To understand the structure of a Block chain and why/when it is better than a simple distributed database
3. To make students understand the technological underpinnings of Block chain operations as distributed data structures and decision making systems

Course Outcomes:

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

1. Define and explain the fundamentals of Block chain.
2. Understand decentralization and the role of Block chain in it.
3. Understand and analyze Bitcioin Crypto currency and underlying Block chain network.
4. Understand Etherium currency and platform, and develop applications using Solidity.
5. Understand Hyper ledger project and its components; critically analyze the challenges and future opportunities in Block chain technology.

UNIT-I:

Introduction: History and basics, Types of Block chain, Consensus, CAP Theorem. Cryptographic Hash Functions: Properties of hash functions, Secure Hash Algorithm, Merkle trees, Patricia trees.

UNIT-II:

Decentralization: Decentralization using Block chain, Methods of decentralization, decentralization framework, Blockchain and full ecosystem decentralization, Smart contracts, Decentralized Organizations, Platforms for decentralization.

UNIT-III:

Bitcoin: Introduction to Bitcoin, Digital keys and addresses, Transactions, Blockchain, The Bitcoin network, Bitcoin payments, Bitcoin Clients and APIs, Alternatives to Proof of Work, Bitcoin limitations.

UNIT-IV:

Etherium: Smart Contracts, Introduction to Ethereum, The Ethereum network, Components of the Ethereum ecosystem, Blocks and Blockchain, Fee schedule, Ethereum Development

Environment, Solidity.

UNIT-V:

Hyperledger: Introduction, Hyperledger Projects, Protocol, Architecture, Hyperledger Fabric, Sawtooth Lake, Corda.

Challenges and Opportunities: Scalability, Privacy, Blockchain for IoT, Emerging trends

Text Book:

1. Mastering Blockchain, Imran Bashir, Second Edition, Packt Publishing.

References:

1. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, Andrea Antonopoulos, and O'Reilly.
2. Blockchain Blueprint for a New Economy, Melanie Swan, O'Reilly.
3. Mastering Bitcoin: Programming the Open Blockchain, Antonopoulos, Andreas M. O'Reilly.
4. Blockchain Technology: Cryptocurrency and Applications, S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan, Oxford University Press.

		L	T	P	C
	Minor Course	**L**	**T**	**P**	**C**
		3	**0**	**2**	**4**
OBJECT ORIENTED PROGRAMMING THROUGH JAVA					

Course Objectives:

1. Implementing programs for user interface and application development using core java principles.
2. Focus on object oriented concepts and java program structure and its installation.
3. Comprehension of java programming constructs, control structures in Java Programming Constructs.

Course Outcomes:

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

1. Understand Java programming concepts and utilize Java Graphical User Interface in Program writing.
2. Write, compile, execute and troubleshoot Java programming for networking concepts.
3. Build Java Application for distributed environment.
4. Design and Develop multi-tier applications.
5. Identify and Analyze Enterprise applications.

UNIT I:

Introduction to OOP - Introduction, Need of Object Oriented Programming, Principles of Object Oriented Languages, Procedural languages Vs OOP, Applications of OOP, History of JAVA, Java Virtual Machine, Java Features, Program Structures, Installation of JDK1.6.

UNIT II:

Variables, Primitive Data types, Identifiers- Naming Conventions, Keywords, Literals, Operators-Binary, Unary and Ternary, Expressions, Precedence rules and Associativity, Primitive Type Conversion and Casting, Flow of Control-Branching, Conditional Loops.

Classes and Objects- Classes, Objects, Creating Objects, Methods, Constructors-Constructor Overloading, Cleaning up Unused Objects-Garbage Collector, Class Variable and Methods-Static Keyword, this keyword, Arrays, Command Line Arguments.

UNIT III:

Inheritance: Types of Inheritance, Deriving Classes using Extends Keyword, Method Overloading, Super Keyword, Final Keyword, Abstract Class.

Interfaces, Packages and Enumeration: Interface-Extending Interface, Interface Vs Abstract Classes, Packages-Creating Packages, Using Packages, Access Protection, java.lang Package.

Exceptions & Assertions - Introduction, Exception Handling Techniques-try...catch, throw, throws, finally block, User Defined Exception, Exception Encapsulation and Enrichment, Assertions.

UNIT IV:

Multi-Threading: java.lang.Thread, The main Thread, Creation of New Threads, Thread Priority, Multithreading- Using isAlive() and join(), Synchronization, Suspending and Resuming Threads, Communication between Threads.

Input/Output: Reading and Writing data, java.io package.

Applet: Applet Class, Applet Structure, Applet Life Cycle, Sample Applet Programs.

UNIT V:

Event Handling: Event Delegation Model, Sources of Event, Event Listeners, Adapter Classes, Inner Classes.

Abstract Window Toolkit : Importance of AWT, Java.awt.package, Components and Containers, Button, Label, Check Box, Radio Buttons, List Boxes, Choice Boxes, Text Field and Text Area, Container Classes, LayOuts, Menu, Scroll bar.

Swings: Introduction, JFrame, JApplet, JPanel, Components in Swings, Layout Managers, List and JScroll Pane, SplitPane, JTabbedPane, JTree, DialogBox, Pluggable Look and Feel.

Text Books:

1. The Complete Reference Java, 8ed, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, Saurabh Choudhary, Oxford.
3. JAVA for Beginners, 4e, Joyce Farrell, Ankit R. Bhavsar, Cengage Learning.

References:

1. JAVA Programming, K.Rajkumar, Pearson.
2. Core JAVA, Black Book, Nageswara Rao, Wiley, Dream Tech.
3. Core JAVA for Beginners, Rashmi Kanta Das, Vikas.
4. Object Oriented Programming through JAVA , P Radha Krishna , University Press.
5. Object oriented programming with JAVA, Essentials and Applications, Raj Kumar Bhuyya,Selvi, Chu TMH.
6. Introduction to Java Programming, 7th ed, Y Daniel Liang, Pearson.

	Minor Course	L	T	P	C
		3	0	2	4
PYTHON PROGRAMMING					

Course Objectives:

1. Introduction to Scripting Language
2. Exposure to various problems solving approaches of computer science
3. **Teach the use of multiple except blocks** to handle different types of exceptions individually and appropriately.

Course Outcomes:

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

1. Understand the fundamentals of scripting language and its learning environment.
2. Acquire the knowledge of data types, operators and control structures.
3. Understand Object oriented concepts and apply the concepts of data structures to real world data.
4. Apply the concept of modularity and implement different packages to solve complex problems. Understand Object oriented concepts and handle different errors through exceptions.
5. Develop multithreaded application using standard libraries.

UNIT-I:

Features and History of Python, Print and Input functions, variables, keywords, comments, **Types:** Numerical Types (int, float, complex), Strings, Boolean, Type Conversion, **Operators:** Arithmetic, Relational, Logical, Bitwise, Assignment, Identity, Membership, **Control Flow:** Indentation, if-elseif-else, while, for, break, continue, pass, else-with loops

UNIT-II:

Functions: Introduction, Required Arguments, Default Arguments, Keyword Arguments, Variable Number of Arguments, Variable Scope and Lifetime, global variables, Lambda Functions, Command Line Arguments

Object Oriented Programming: Classes and Objects, built-in class methods and attributes, ‘self’, constructor, destructor, inheritance, data hiding, overriding methods and overloading operators

UNIT-III:

Data Structures: Lists, Nested Lists, List Comprehensions, Tuples and Sequences, Sets, Dictionaries

File I/O: opening, closing, reading and writing.

UNIT-IV:

Exception Handling: Exceptions, Multiple Except Blocks, Multiple Exceptions in a Single Block, Except Block without Exception, the else Clause, Raising Exceptions, Built-in and User-defined Exceptions, The finally block, Introduction modules, import and from-import, Packages in Python, used defined modules and packages, PIP.

UNIT-V:

The Python Standard Library: numeric and mathematical modules, string processing, date & time, calendar, operating system, web browser

Graphics with turtle: Motion Control, Pen, Colour, Fill, multiple turtles, reset and clear

GUI design with tkinter: Button, Canvas, Check button, Entry, Frame, Label, Listbox, Menu, Menu button, Message, Radio button, Scale, Scrollbar, Text

Text Books:

1. Python Programming using problem solving approach, Reema Thareja, Oxford University Press.
2. Learning Python, Mark Lutz, O'Rielly
3. Programming Python, Fourth Edition, Mark Lutz, O'Reilly Media.

References:

1. Introduction to Computation and Programming Using Python with Application to Understanding, John V. Guttag, PHI.
2. Think Python: How to think like a Computer Scientist, Allen Downey, Green Tea Press.
3. Head First Python: A Brain-Friendly Guide, Second Edition, Paul Barry, O'Reilly
4. The Python Standard Library, Python 3.6.5 documentation (Web Resource) https://docs.python.org/3/library/

	Minor Course	L	T	P	C
		3	0	2	4
BASIC WEB DESIGNING					

Course Objectives:

The objectives of this course are to acquire knowledge on the

1. How does a website work and web related terminology.
2. Web standards and W3C elements
3. Responsive Web Designing

Course Outcomes:

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

1. Learn the basic terminology related to web and web development.
2. Learn how to design static web pages by using HTML.
3. Should be able to create web pages with enhanced look and feel by Using CSS.
4. Learn to use Java Script for design thick clients and to design interactive responsive form design and validations.
5. Learn to design and host and publish websites in various domains.

UNIT - I:

Introduction to Web and Web Design Principles: Brief History of Internet, What is World Wide Web, Why create a web site, Web Standards, Web pages, Website, Web browsers and Web servers and Web protocols.

Basic principles involved in developing a web site, Planning process, Five Golden rules of web designing ,Designing navigation bar , Page design ,Home Page Layout ,Design concept.

UNIT - II:

Introduction to HTML: What is HTML, HTML Documents, Basic structure of an HTML document, Creating an HTML document, Mark up Tags, Heading-Paragraphs, Line Breaks, HTML Tags. Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia ,Working with Forms and controls.

UNIT - III:

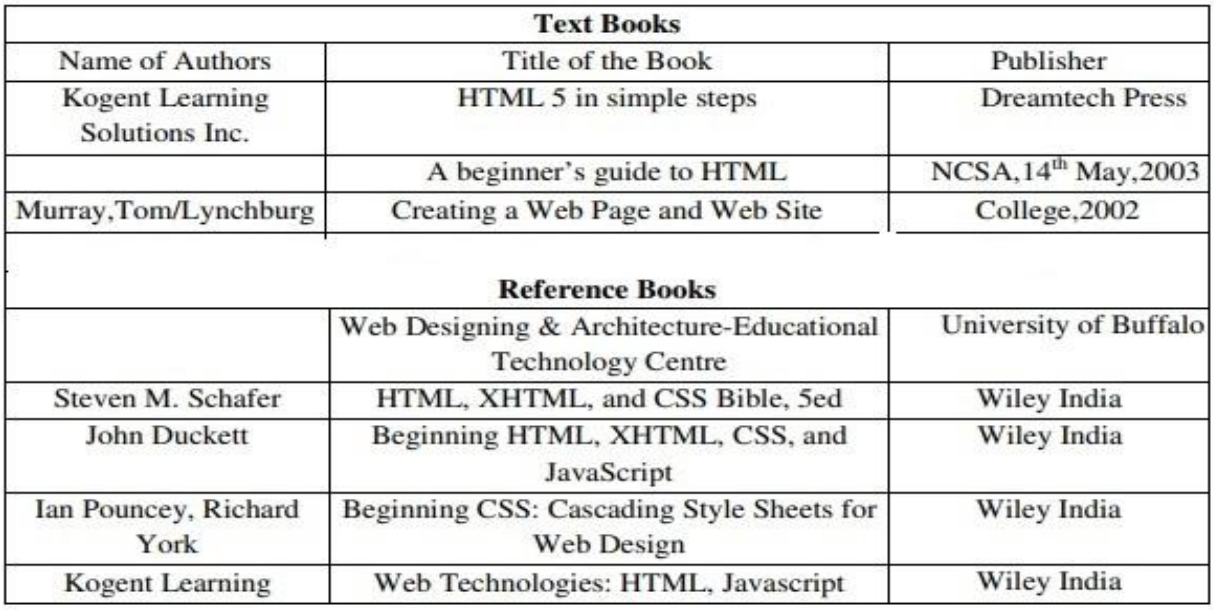
Introduction to Cascading Style Sheets: Concept of CSS, Creating Style Sheet, CSS Properties, CSS Styling(Background, Text Format, Controlling Fonts), Working with block elements and objects, working with Lists and Tables, CSS Id and Class ,Box Model(Introduction, Border properties, Padding Properties, Margin properties) ,CSS Advanced(Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute sector) , CSS Color ,Creating page Layout and Site Designs.

UNIT - IV:

Introduction to Java Script: What is Java Script? Basics of Java Script: Variables, functions, and Operators, select HTML elements with Java Script, Java Script Events and Event Handlers, Regular expressions and pattern matching in Java Script. Form validation using Java Script.

UNIT - V:

Introduction to Web Publishing or Hosting: Creating the Web Site, Saving the site, working on the web site, Creating web site structure, Creating Titles for web pages, Themes-Publishing web sites. Case study: Web publishing and hosting using Heroku cloud platform (https://www.heroku.com/).



Text Books		
Name of Authors	Title of the Book	Publisher
Kogent Learning Solutions Inc.	HTML 5 in simple steps	Dreamtech Press
	A beginner's guide to HTML	NCSA,14th May,2003
Murray,Tom/Lynchburg	Creating a Web Page and Web Site	College,2002
Reference Books		
	Web Designing & Architecture-Educational Technology Centre	University of Buffalo
Steven M. Schafer	HTML, XHTML, and CSS Bible, 5ed	Wiley India
John Duckett	Beginning HTML, XHTML, CSS, and JavaScript	Wiley India
Ian Pouncey, Richard York	Beginning CSS: Cascading Style Sheets for Web Design	Wiley India
Kogent Learning	Web Technologies: HTML, Javascript	Wiley India

	Minor Course	L	T	P	C
		3	0	2	4
ADVANCED WEB TECHNOLOGIES					

Course Objectives:

The objectives of this course is to acquire knowledge on the

1. This course is designed to introduce students with basic web programming experience to the advanced web programming languages and techniques associated with the World Wide Web.
2. The course will introduce web-based media-rich programming tools for creating interactive web pages.
3. The course will introduce Web Frame works like React JS and Angular JS for quick and efficient design and implementation of web applications.

Course Outcomes:

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

1. Analyze a web page and Create web pages using HTML5 and Cascading Styles sheets 3 and Boot strap.
2. Build dynamic web pages using Java Script and Write simple client-side scripts using AJAX.
3. Learn to use XML for data exchange and transfer over web and XML parsing and validation techniques.
4. Build web applications using PHP.
5. Describe a java web services.

UNIT – I

Introduction to HTML5, CSS3 and Boot strap: Basic Syntax, Standard HTML Document Structure, HTML5 tags, Audio, video, 2D canvas Drawing and animations using HTML5.

CSS 3: What is SCSS, Difference between CSS and SCSS, Introduction to SASS tool and CSS template design using Bootstrap

UNIT - II:

Java Script and DHTML: DHTML: Java Script DOM, Interactive and responsive web page designing, Positioning Moving and Changing Elements.

Java Script Web Frame works: React JS, Angular JS and Vue JS, Single Page Application (SPA) Design and Development using Angular JS.

UNIT - III:

XML: Introduction to XML, XML vs HTML, Document type Definition, XML schemas, Document object model, XSLT, DOM and SAX Approaches. AJAX A New Approach: Introduction to AJAX. Request and Response mechanism of AJAX.

UNIT - IV:

PHP Programming: Introducing PHP: Creating PHP script, Running PHP script. Working with variables and constants: Using variables, Using constants, Datatypes, Operators. Controlling program flow: Conditional statements, Control statements, Arrays, functions. Working with forms and Databases such as MySQL.

UNIT - V:

Web Services: JAX-RPC-Concepts-Writing a Java Web Service-Writing a Java Web Service Client- Describing Web Services: WSDL- Representing Data Types: XML Schema Communicating Object Data: SOAP Related Technologies-Software Installation-Storing Java Objects as Files-Databases and Java Servlets.

Text Books:

1. Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
2. Introducing HTML5 (Voices That Matter) 2nd Edition by Bruce Lawson / Remy Sharp Lawson / Sharp,Kindle publishers.
3. Web Technologies, Uttam K Roy, Oxford
4. HTML, CSS, and JavaScript All in One: Covering HTML5, CSS3, and ES6, Sams Teach Yourself 3rd Edition, by Julie Meloni and, Jennifer Kyrnin.Pearson
5. JavaScript Frameworks for Modern Web Development: The Essential Frameworks, Libraries, and Tools to Learn Right Now 2nd ed. Edition by Sufyan bin Uzayr , Nicholas Cloud , Tim Ambler.Apress.
6. Java Web Services: Up and Running: A Quick, Practical, and Thorough Introduction 2ndEdition, Kindle Edition by Martin Kalin.

References:

1. Ruby on Rails Up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly(2006
2. Programming Perl, 4ed, Tom Christiansen, Jonathan Orwant, Oreilly (2012)
3. Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
4. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning.

	Minor Course	L	T	P	C
		3	1	0	4
COMPUTER ORGANIZATION AND ARCHITECTURE					

Course Objectives:

1. To understand the structure, function and characteristics of computer system.
2. To understand the design of the various functional units and components of computers.
3. To explain the function of each element of a memory hierarchy.

Course Outcomes:

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

1. Understand the architecture of modern computer.
2. Analyze the Performance of a computer using performance equation.
3. Understand different instruction types.
4. Calculate the effective address of an operand by addressing modes.
5. Understand how computer stores positive and negative numbers.
6. Understand how computer performs arithmetic operation of positive and negative numbers.

UNIT -I:

Basic Structure of Computers: Functional unit, Basic Operational Concepts, Bus Structures, System Software, Performance, The History of Computer Development. Data Representation: Data Types, Complements, Fixed Point Representation, Floating Point Representation.

Basic Computer Organization and Design: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory – Reference Instructions, Interrupt, Design of Basic Computer, Design of Accumulator Logic.

UNIT -II:

Machine Instruction and Programs: Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Addressing Modes, Basic Input/output Operations, Importance of Stacks and Queues in Computer Programming Equation. Component of Instructions: Logic Instructions, Shift and Rotate Instructions, Branch Instructions.

Computer Arithmetic: Addition, Subtraction, Multiplication and Division Algorithms. Floating point Arithmetic Operations and Decimal Arithmetic Operations.

UNIT -III:

The Memory System: Memory System Consideration RAM and ROM, Flash Memory, Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory and Virtual Memory, Secondary Storage: Magnetic Hard Disks, Optical Disks.

Pipeline Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline.

UNIT -IV:

Processing Unit: Fundamental Concepts: Register Transfers, Performing an Arithmetic or Logic Operation, Fetching a Word from Memory, Execution of Complete Instruction, Hardwired Control.

Micro Programmed Control: Microinstructions, Micro Program Sequencing, Wide Branch Addressing and Microinstructions with Next – Address Field.

UNIT -V:

Input / Output Organization:

Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB).

Parallelism:

Instruction-Level-Parallelism – Parallel Processing Challenges – Flynn's Classification: SISD, MIMD, SIMD, SPMD and Vector Architectures, – Hardware Multithreading – Multi-Core Processors and Other Shared Memory Multiprocessors.

Text Books:

1. Computer Organization, Carl Hamacher, ZvonksVranesic, SafeaZaky, 6th Edition, McGraw Hill.
2. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill.

References:

1. Computer Organization and Architecture – William Stallings tenth Edition, Pearson/PHI.
2. Computer System Architecture, M. Morris Mano, 3 rd Edition Pearson Education.
3. Structured Computer Organization – Andrew S.Tanenbaum, 4th Edition PHI/Pearson.
4. Fundamentals of Computer Organization and Design, Sivarama Dandamudi Springer Int.Edition.

		L	T	P	C
	Minor Course	3	1	0	4
DISTRIBUTED SYSTEMS					

Course Objectives:

1. To understand the foundations of distributed systems.
2. To learn issues related to clock Synchronization, the need for global state and remote invocation in distributed systems.
3. To learn distributed mutual exclusion and deadlock detection algorithms.

Course Outcomes:

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

1. Understand the foundations and issues of distributed systems.
2. Illustrate the various synchronization issues, global state and remote invocation for distributed systems.
3. Develop the Mutual Exclusion and Deadlock detection algorithms in distributed systems.
4. Apply the features of peer-to-peer, distributed shared memory systems and security.
5. Analyze the distributed transactions, agreement protocols and fault tolerance mechanisms in distributed systems.

UNIT- I:

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges, Relation to Computer system Components, Motivation, Relation to Parallel Systems, Message-Passing systems versus Shared Memory systems, Primitives for Distributed Communication, Synchronous versus Asynchronous executions, Design issues and Challenges.

UNIT –II:

Message Ordering and Group Communication: Message ordering paradigms, Asynchronous execution with synchronous communication, Synchronous program order on an asynchronous system, Group communication, Causal order (CO), Total order.

Global state and Snapshot Recording Algorithms: Introduction, System model and definitions, Snapshot algorithms for FIFO channels. Remote Invocation: Introduction, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection, Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI.

UNIT- III:

Distributed Mutual Exclusion Algorithms: Introduction, Preliminaries, Lamport's algorithm, Ricart-Agrawala algorithm, Maekawa's algorithm, Suzuki–Kasami's broadcast algorithm.

Deadlock Detection in Distributed Systems: Introduction, System model, Preliminaries, Models of deadlocks, Knapp's Classification, Algorithms for the Single Resource Model, the AND model and the OR model.

UNIT -IV:

Peer-to-Peer Computing and Overlay Graphs: Introduction, Data indexing and overlays, Chord distributed hash table, Content addressable networks, Tapestry.

Distributed Shared Memory: Abstraction and advantages, Memory consistency models, Shared Memory Mutual Exclusion.

Security: Introduction, Overview of Security Techniques, Cryptographic Algorithms, Digital Signatures, Cryptography Pragmatics.

UNIT –V:

Distributed Transactions: Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions. Check Pointing and **Rollback Recovery:** Introduction, Background and definitions, Issues in Failure recovery, Checkpoint-based recovery, Log-based rollback recovery, coordinated check pointing algorithm, Algorithms for asynchronous and synchronous check pointing and recovery. **Consensus and Agreement Algorithms:** Problem definition, Overview of results, Agreement in a Failure-Free system (synchronous or asynchronous).

Text Books:

1. Distributed computing: Principles, algorithms, and systems, Ajay D Kshemkalyani and Mukesh Singhal, Cambridge University Press, 2011.
2. Distributed Systems Concepts and Design, George Coulouris, Jean Dollimore and TimKindberg, 5th Edition, Pearson Education, 2012.

References:

1. Distributed Operating Systems: Concepts and Design, Pradeep K Sinha, Prentice Hall ofIndia, 2007.
2. Advanced concepts in operating systems. Mukesh Singhal and Niranjan G. Shivaratri,McGraw-Hill, 1994.
3. Distributed Systems: Principles and Paradigms, Tanenbaum A.S., Van Steen M.,Pearson Education, 2007.

E-Resources:

1. https://nptel.ac.in/courses/106/106/106106168/

	Minor Course	**L**	**T**	**P**	**C**
		3	**1**	**0**	**4**

CLOUD COMPUTING

Course Objective:

1. Explain the evolution of computing paradigms,
2. Differentiate cloud computing as both a service and a platform
3. Identify and evaluate different types of applications suitable for cloud deployment,

Course Outcomes:

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

1. Understand and analyze different computing paradigms
2. Understand the basics of cloud computing and different cloud deployment models.
3. Understand different cloud implementation and management strategies.
4. Understand and evaluate different cloud service models.
5. Identify, analyze and use different cloud services/applications/tools available from key cloud providers.

UNIT-I:

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing.

UNIT-II:

Cloud Computing Fundamentals: Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud Computing, Cloud Computing is a Service, Cloud Computing is a Platform, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models

UNIT-III:

Cloud Computing Architecture and Management: Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure, Managing the Cloud Application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

UNIT-IV:

Cloud Service Models: Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platformas a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models.

UNIT-V:

Cloud Providers and Applications: EMC, EMC IT, Captiva Cloud Toolkit, Google Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA

Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rackspace, VMware, Manjra soft, Aneka Platform.

Text Book:

1. Essentials of Cloud Computing, K. Chandrasekhran, CRC press.

References:

1. Cloud Computing: Principles and Paradigms, Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, SubraKumaraswamy, ShahedLatif, O'Reilly.

	Minor Course	**L**	**T**	**P**	**C**
		3	**1**	**0**	**4**

QUANTUM COMPUTING

Course Objectives

1. To introduce the fundamentals of quantum computing
2. The problem-solving approach using finite dimensional mathematics
3. Explain the principle of superposition and the concept of entanglement, highlighting their significance in quantum mechanics and computation.

Course Outcome

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

1. Understand the Basics of complex vector spaces
2. Quantum mechanics as applied in Quantum computing
3. Apply Quantum Architecture and algorithms to solve real time problems.
4. Understand and explore the models of Quantum Computer and Quantum Simulation tools
5. Analyze and implement basic quantum algorithms involving superposition, entanglement, and measurement operations.

UNIT-I:

Introduction: Complex numbers and its geometrical representations, Complex vector spaces, inner products and Hilbert spaces, Hermitian and unitary matrices, Tensor products of vector spaces Deterministic Systems

UNIT-II:

Dirac formalism, superposition of states, entanglement Bits and Qubits. Qubit operations, Hadamard Gate, CNOT Gate, Phase Gate, Z-Y decomposition, Quantum Circuit Composition, Basic Quantum circuits.

UNIT-III:

Quantum Algorithm- I: Quantum parallelism, Quantum Evolution, Deutsch's Algorithm, Deutsch-Jozsa Algorithm, Simon's periodicity algorithm.

UNIT-IV:

Quantum Algorithm- II: Grover's search algorithm, Shor's Factoring algorithm. Application of entanglement, teleportation, superdence coding.

UNIT-V

Quantum Software Development and Programming:

Quantum programming languages, Probabilistic and Quantum computations, introduction to quantum cryptography and quantum information theory.

Text Books

i. Quantum computing explained, David McMahon, Wiley-interscience, John Wiley &

Sons, 2008

ii. Quantum computing for computer scientists, Noson S. Yanofsky, Mirco A. Mannucci, Cambridge University Press 2008

Reference Books

i. Quantum computation and quantum information, Michael A. Nielsen and Isaac L. Chuang, Cambridge University Press 2010

ii. Introduction to Quantum Mechanics, 2nd Edition, David J. Griffiths, Prentice Hall New Jersey 1995