Jharkhand University of Technology, Ranchi
NEP-2020 based Syllabus w.e.f – 2025-26 Batch
B.Tech in Computer Science and Engineering

Semester-Vth

S.No.	Course Code	Course Title	L	T	P	J	Cr	FM	Overall Pass Marks	Internal	External	Categorization
			Contact Hours per week									
		PROGRAMME CORE COURSES (PCC)										
01		PCC-I (Software Engineering)	3	0	0		3	100	35	30	70	
02		PCC-II (Cloud Computing)	3	0	0		3					
		***PROGRAMME ELECTIVES I & II**										
03		PE-I (Any one from the given basket of PE-I)										
04		PE-II (Any one from the given basket of PE- II)										
05		Entrepreneurship Management										
Total												
Practical			**L**	**T**	**P**	6	**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**
06		Seminar & Report Writing –I (UN Sustainable Development Goals)										
07		Business Communications										

08		**Lab-I** (Computer Network LAB)	0	0	3		1					
09		**Lab-II** (Software Engineering LAB)	0	0	3		1					
Total												
Audit Course			**L**	**T**	**P**		**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**
10		**REGIONAL LANGUAGE (ANY ONE THROUGH NPTEL / SWAYAM)** 1. MARATHI 2. KANNAD 3. TAMIL 4. TELUGU										
11		Sports/NCC/NSS/YOGA/Painting/Music/ Classical Dance										
Total												
Grand Total												

Basket of Professional Elective-I

S. No.	Course Code	Subject
01		Advanced Algorithms
02		Advanced Computer Architecture
03		Artificial Intelligence
04		Computational Number Theory
05		Programming Language Concepts
06		Software Testing
07		Computational Logic
08		Neuro Fuzzy and Genetic Programming
09		Digital Image Processing
10		Network Security

Basket of Professional Elective-II

S. No.	Course Code	Subject
01		Data Mining and Analytics
02		Distributed Operating Systems
03		Pattern Recognition Techniques
04		Biometrics
05		Information Storage and Management
06		Wireless Sensor Networks
07		Network Protocols and Programming
08		Network Routing Algorithms
09		High Performance Computing
10		Cyber Security Essentials for AI & IoT

Abbreviations:-*AU- Audit Course; L: Lecture, T: Tutorial, P: Practice.
J- Self learning hours shall not be reflected in the Time table. Self-learning includes micro project/ assignment/ other activities as mentioned in earlier semester.
***Passing in Audit Course shall be mandatory.**
Note:- Student may choose their two Professional Electives (PE-I & PE-II) from NPTEL/SWAYAM also on the advice of departmental academic council if the subject is not mentioned in the above basket.
Students will complete the Elective Papers (Professional) of 12 weeks duration from NPTEL/SWAYAM. Student may register on NPTEL/SWAYAM at any time from 1st to 5th semester but the passing marks and credits will be reflected only in the 5th semester.
The secured percentage of marks and passing certificate of the subject shall be forwarded by the institute to Controller of Examination (CoE), JUT, Ranchi timely.

Jharkhand University of Technology, Ranchi
NEP-2020 based Syllabus w.e.f – 2025-26 batch
B.Tech in CSE

Semester- VIth

S.No.	Course Code	Course Title	L	T	P	J	Cr	FM	Overall Pass Marks	Internal	External	Categorization
			Contact Hours per week									
		PROGRAMME CORE COURSES (PCC)										
01		PCC-I (Compiler Design)	3	0	0	3						
02		PCC-II (AI and Machine Learning)	3	0	0	3						
		PROGRAMME ELECTIVES III & IV										
03		PE-III (Any One FromThe Given Basket of PE-III & IV)										
04		PE-IV (Any One FromThe Given Basket of PE-III & IV)										
		OPEN ELECTIVE-I										
05		OE-I (Any one FromThe Given Basket of OE-I)										
Total												
Practical			**L**	**T**	**P**		**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**
06		**Lab-I (Compiler Design Lab)**			**3**		**1**					

07		**Lab-II (Machine Learning LAB)**			**3**		**1**					
08		**Lab-III (Cyber Security LAB)**		**1**	**2**		**2**					
Total												
Audit Course			**L**	**T**	**P**		**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**
09		**FOREIGN LANGUAGE (ANY ONE THROUGH NPTEL / SWAYAM)** 1. GERMAN 2. JAPANESE 3. CHINESE 4. KOREAN 5. SPAINISH 6. RUSSIAN 7. FRENCH										
10		Sports/NCC/NSS/YOGA/Painting/Music/ Classical Dance										
Internship			**L**	**T**	**P**		**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**
11		Industrial Internship	08 weeks									
Minor Project (For Exit Option)												
Total												

Grand Total										

Basket of Professional Elective- III

S. No.	Course Code	Subject
01		Database Security and Privacy
02		Software Defined Networks
03		Semantic Web
04		Wireless and Mobile Communication
05		Service Oriented Architecture
06		Network Design and Management
07		Advanced Data Mining
08		Advanced Databases
09		Bio-Informatics
10		Computer Vision and Image Processing

Basket of Professional Elective- IV

S. No.	Course Code	Subject
01		Data Networks
02		Design Patterns
03		Foundations of Blockchain Technology
04		Information Retrieval
05		Mobile Computing
06		Natural Language Processing
07		Program Analysis and Verification
08		Software Metrics and Software Project Management
09		Distributed Computing
10		Introduction to Generative AI

Basket of Open Elective I

S. No.	Course Code	Subject	S. No.	Course Code	Subject
01		IT Infrastructure Management	11		Drinking Water Supply and Treatment
02		Mobile Application Development	12		Electric Vehicle Technology
03		System Modeling and Simulation	13		Introduction to PLC Programming
04		Free and Open Source Softwares	14		Nano Technology
05		Android Development	15		Functional Materials
06		Data Analysis using Open-Source Tool	16		Traditional Indian Foods
07		Concept of Industrial Electronics	17		Introduction to food processing
08		Fundamentals of Aeronautical Engineering	18		IPR for Pharma Industry
09		Remote Sensing Concepts	19		Basics of Textile Finishing
10		Urban Agriculture	20		Database Management system

Abbreviations:- *AU- Audit Course; L: Lecture, T: Tutorial, P: Practice.

J- Self learning hours shall not be reflected in the Time table. Self-learning includes micro project/ assignment/ other activities as mentioned in earlier semester.

***Passing in Audit Course shall be mandatory.**

Note:- Student may choose their two Professional Electives (PE-III & PE-IV)& Open Elective-I from NPTEL/SWAYAM also on the advice of departmental academic council if the subject is not mentioned in the above basket.

Students will complete the Elective Papers (Professional or Open) of 12 weeks duration from NPTEL/SWAYAM. Student may register on NPTEL/SWAYAM at any time from 1st to 6th semester but the passing marks and credits will be reflected only in the 6th semester.

The secured percentage of marks and passing certificate of the subject shall be forwarded by the institute to Controller of Examination (CoE), JUT, Ranchi timely.

Jharkhand University of Technology, Ranchi
NEP-2020 based Syllabus w.e.f – 2025-26 batch
B.Tech in CSE

Semester- VIIth

S.No.	Course Code	Course Title	L	T	P	J	Cr	FM	Overall Pass Marks	Internal	External	Categorization
			Contact Hours per week									
		PROGRAMME CORE COURSES (PCC)										
01		PCC-I (Data Science)	3	0	0		3					
		PROGRAMME ELECTIVES V, VI & VII										
02		PE-V (Any One FromThe Given Basket of PE-V)										
03		PE-VI (Any One FromThe Given Basket of PE-VI)										
04		PE-VII (Any One FromThe Given Basket of PE-VII)										
		OPEN ELECTIVE-II & III										
05		OE-II (Any one From The Given Basket of OE-II)										
06		OE-III (Any one FromThe Given Basket of OE-III)										
Total												
Practical			**L**	**T**	**P**		**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**

06		**Lab-I (Data Science LAB)**	**0**	**0**	**3**		**1**					
07		**Lab-II (Digital Project Management LAB)**	**0**	**1**	**2**		**2**					
Total												
Audit Course			**L**	**T**	**P**		**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**
10		**(THROUGH NPTEL/SWAYAM)** Exploring Human Values: Visions of Happiness and Perfect Society										
11		Sports/NCC/NSS/YOGA/Painting/Music/ Classical Dance										
Project			**L**	**T**	**P**		**Cr**	**FM**	**Overall Pass Marks**	**Internal**	**External**	**Categorization**
12		Minor Research Project										
Total												
Grand Total												

Basket of Professional Elective-V

S. No.	Course Code	Subject
01		Cloud Computing
02		Deep Learning
03		Design of Secure Protocols

Basket of Professional Elective-VI

S. No.	Course Code	Subject
01		AR/VR Systems for Real-World Applications
02		Software Reliability Techniques
03		Pattern Recognition

04		Internet of Things		**04**		Soft Computing
05		Intrusion Detection Systems		**05**		Multimedia Technology
06		Quantum Computing		**06**		Time Series Analysis and Forecasting (TSAF)
07		Real Time Systems		**07**		Image Processing
08		Secure Software Engineering		**08**		Digital Signal Processing
09		Security and Privacy		**09**		Video Analytics
10		Robotic Process Automation		**10**		Social Network Analysis

Basket of Professional Elective-VII

S. No.	Course Code	Subject		S. No.	Course Code	Subject
01		Big Data Analytics		06		Optimization Techniques
02		Information Storage and Retrieval		07		Information and Coding Theory
03		Programming Environment and User Interface Design		08		Cryptography
04		Advanced Numerical Computation		09		Information Security
05				10		Approximation Algorithms

Basket of Open Elective-II

S. No.	Course Code	Subject
01		Advanced Numerical Methods
02		Random Processes
03		Queuing and Reliability Modelling
04		Production and Operations Management for Entrepreneurs
05		Multivariate Data Analysis
06		Additive Manufacturing
07		New Product Development
08		Industrial Design & Rapid Prototyping

Basket of Open Elective-III

S. No.	Course Code	Subject
01		Reverse Engineering
02		Sustainable Manufacturing
03		Electric and Hybrid Vehicles
04		Space Engineering
05		Industrial Management
06		Quality Engineering
07		Fire Safety Engineering
08		Introduction to Non-destructive Testing

		Techniques				
09		Micro and Precision Engineering		**09**		Mechatronics
10		Cost Management of Engineering Projects		**10**		Foundation of Robotics

Abbreviations:- *AU- Audit Course; L: Lecture, T: Tutorial, P: Practice.
J- Self learning hours shall not be reflected in the Time table. Self-learning includes micro project/ assignment/ other activities as mentioned in earlier semester.
***Passing in Audit Course shall be mandatory.**
Note:- Student may choose their two Professional Electives (PE-V, VI & PE-VII) & Open Elective-II & III from NPTEL/SWAYAM also on the advice of departmental academic council if the subject is not mentioned in the above basket.
Students will complete the Elective Papers (Professional or Open) of 12 weeks duration from NPTEL/SWAYAM. Student may register on NPTEL/SWAYAM at any time from 1st to 7th semester but the passing marks and credits will be reflected only in the 7th semester.
The secured percentage of marks and passing certificate of the subject shall be forwarded by the institute to Controller of Examination (CoE), JUT, Ranchi timely.

Jharkhand University of Technology, Ranchi
NEP-2020 based Syllabus w.e.f – 2025-26 batch
B.Tech, Branch-

VIIIth –Semester

S.No.	Course Code	Course Title	L	T	P	J	Cr	FM	Overall Pass Marks	Internal	External	Categorization
			Contact Hours per week									
01		Major Project/ Research Project	36 Hours per week / week Total 12-16 Weeks			6	20					
02		Industrial Internship										
Total												
Grand Total												

L: Lecture, T: Tutorial, P: Practical.
J- Self learning hours shall not be reflected in the Time table. Self-learning includes micro project/ assignment/ other activities as mentioned in earlier semester.

Jharkhand University of Technology, Ranchi

B.Tech
Computer Science and Engineering

NEP-2020 based Syllabus w.e.f – 2025-26 Batch

Semester- Vth

Software Engineering

Course Code: **L:T:P: 3:0:0**

Rationale:
Software Engineering forms the backbone of computer science education and professional practice, providing structured approaches to the design, development, testing, and maintenance of software systems. This course introduces students to the Software Development Lifecycle (SDLC), core concepts, and fundamental practices of software engineering while building their problem-solving and programming skills. By exploring the relationships between software engineering and other engineering disciplines, students gain an interdisciplinary perspective essential for large-scale, real-world projects. The course emphasizes both theory and hands-on practice, enabling students to work with development environments, troubleshoot and debug programs, and build original applications using Python. By laying a strong conceptual and practical foundation, the course prepares students for advanced coursework and fosters the skills necessary for academic and professional success in the field of Software Engineering.

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

CO1: Explain the role of software engineers, software characteristics, and various software development life cycle (SDLC) models in addressing software crises.
CO2: Apply requirement engineering processes including elicitation, analysis, modeling, and documentation to develop a Software Requirement Specification (SRS) document for real-world problems.
CO3: Evaluate software quality concepts and prepare Software Quality Assurance (SQA) plans by applying established quality frameworks and standards (ISO, CMM).
CO4: Analyze and implement various testing strategies (unit, integration, regression, functional, structural) for conventional, object-oriented, and web-based applications.
CO5: Demonstrate project management skills by estimating resources, scheduling tasks, managing risks, and planning for maintenance and re-engineering of software projects.

Course Contents:

UNIT I
Introduction: Role of Software Engineer – Software Components – Software Characteristics – Software Crisis – Software Engineering Processes – Similarity and Differences from Conventional Engineering Processes – Quality Attributes.
Assessment: How Software Engineering Changes? Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Choosing a social relevant problem, Summary Team Report.

UNIT II
Requirement Engineering Process: Elicitation – Analysis – Documentation – Review and Management of User Needs – Feasibility Study – Information Modeling – Data Flow Diagrams – Entity Relationship Diagrams – Designing the architecture.
Assessment: Impact of Requirement Engineering in their problem, Decision Tables, SRS Document, IEEE Standards for SRS, Architectural design, component level design, user interface design, WebApp Design, Submission of SRS Document for Team Project.

UNIT III

Quality concepts – Review techniques – Software Quality Assurance (SQA): Verification and Validation – SQA Plans – Software Quality Frameworks. Assessment: Framing SQA Plan, ISO 9000 Models, SEI-CMM Model and their relevance to project Management, other emerging models like People CMM.

UNIT IV

Testing Objectives – Unit Testing – Integration Testing – Acceptance Testing – Regression Testing – Testing for Functionality and Testing for Performance – Top-Down and Bottom-Up Testing – Software Testing Strategies – Strategies: Test Drivers and Test Stubs – Structural Testing (White Box Testing) – Functional Testing (Black Box Testing) – Testing conventional applications – object oriented applications – Web applications – Formal modeling and verification – Software configuration management – Product metrics. Assessment: Team Analysis in Metrics Calculation.

UNIT V

Project Management Concepts – Process and Project Metrics – Estimation for Software projects – Project Scheduling – Risk Management – Maintenance and Re-engineering. Assessment: Preparation of Risk mitigation plan.

Learning Resources

Text Books

1. R. S. Pressman, “Software Engineering: A Practitioners Approach”, Seventh Edition, McGraw Hill, 2010.
2. Rajib Mall, “Fundamentals of Software Engineering”, Third Edition, PHI Publication, 2009.
3. Pankaj Jalote, “Software Project Management in Practice”, Pearson Education, New Delhi, 2002.

Cloud Computing

Course code: **L:T:P:3:0:0**

Rationale:

Cloud Computing has revolutionized the way computing resources are delivered, managed, and consumed. This course introduces students to the fundamentals of cloud computing, including cloud architecture, services, virtualization, and security challenges. It emphasizes the practical aspects of cloud adoption, cost analysis, and deployment models while addressing enterprise-level applications and case studies from leading providers like Amazon, Microsoft, and Google. By the end of the course, students will gain both conceptual knowledge and practical insights into leveraging cloud technologies for modern business and research needs.

Course Outcomes:

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

CO1: Describe the fundamentals, types, characteristics, and value propositions of cloud computing, including cost models and service-level agreements.
CO2: Analyze cloud architecture and service models (IaaS, PaaS, SaaS) and explain how they enable scalable and flexible IT solutions.
CO3: Evaluate cloud security challenges and apply identity management protocols to ensure secure access to cloud environments.
CO4: Demonstrate understanding of virtualization technologies, hypervisors, load balancing, and capacity planning in cloud infrastructure.
CO5: Apply cloud computing concepts to real-world applications through case studies of enterprise data centers, web services, and multi-cloud environments.

Course Contents:

UNIT - I Fundamentals of Cloud Computing **9 Hour**
Define Cloud Computing, Cloud Types, Characteristics of Cloud Computing, Benefits and disadvantages of cloud systems, Assessing the Value Proposition, Measuring the Cloud's Value, Capital Expenditures, Total Cost of Ownership, Service Level Agreements, Licensing Models

UNIT - II Cloud Architecture and Services **9 Hour**
Cloud Computing Stack, Composability, Infrastructure, Platforms and Virtual Appliances, Communication Protocols and Applications, Connecting to the Cloud, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS)

UNIT - III Cloud Security **9 Hour**
Cloud Security Challenges, Software-as-a-Service Security, End-User Access to Cloud Computing Overview, Identity Protocol Standards, Windows Azure Identity Standards

UNIT - IV Virtualization and Capacity Planning **9 Hour**

Virtualization Technologies, Abstraction versus Virtualization, Load Balancing and Virtualization, The Google Cloud, Hypervisors, Virtual Machine Imaging, Porting Applications, Capacity Planning

UNIT - V Cloud Computing Applications **9 Hour**

Web Services: Amazon, Microsoft, Google, Case Studies: Cloud as Infrastructure for an Internet Data Center (IDC), Cloud Computing for Software Parks, Enterprise with Multiple Data Centers

Learning Resources:

Text Books:

1. Barrie Sosinsky (2011), *Cloud Computing Bible*, Wiley Publishing Inc.
2. John W. Rittinghouse and James F. Ransome (2010), *Cloud Computing, Implementation, Management, and Security*, CRC Press.
3. Sunil Kumar Manvi, Gopal K. Shyam (2021), *Cloud Computing: Concepts and Technologies*, CRC Press, 1st Edition.

References:

1. Borko Furht, Armando Escalante (2010), *Handbook of Cloud Computing*, Springer.
2. Michael Kavis (2014), *Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS)*, John Wiley & Sons.

***PROGRAMME ELECTIVES I & II**

PE-I (Any One FromThe Given Of PE-I)

Advanced Algorithms

Course code: **L:T:P: 3:0:0**

Rationale:

Advanced Algorithms is a core subject that equips students with the analytical and problem-solving skills required for tackling complex computational challenges. The course explores algorithm design techniques such as dynamic programming, greedy methods, graph and network flow algorithms, randomized and approximation algorithms, and advanced mathematical models like linear programming. It emphasizes both theoretical foundations and practical applications, including string matching, sorting networks, matrix operations, and geometric algorithms. By engaging with case studies and modern algorithmic approaches, students will develop the ability to analyze algorithmic efficiency, apply advanced problem-solving strategies, and address real-world computational problems in diverse domains.

Course Outcomes:

CO1: Apply advanced techniques such as dynamic programming, divide-and-conquer, and probabilistic analysis to solve computational problems efficiently.

CO2: Analyze and implement graph algorithms, network flow methods, and parallel algorithms for real-world applications.

CO3: Evaluate the performance of sorting networks, matrix algorithms, and linear algebraic computations in terms of time and space complexity.

CO4: Design and analyze string matching and randomized algorithms, and assess their computational complexity and applicability.

CO5: Examine NP-completeness, develop approximation algorithms, and apply linear programming models to solve optimization and geometric problems.

Course Contents:

Unit-1 – Introduction to Algorithm Analysis **9 Hour**

Role of algorithms in computation, Notation order, Recurrences, Probabilistic Analysis, Statistics of Sorting, and its order notations: Heap Sort and Quick Sort in Linear time and space, Dynamic Programming: Matrix Chain Multiplication, Operations on Polynomials, DFT & FFT Algorithms, longest common Subsequence, and optimal binary Search trees

Unit-2 – Graph Algorithms and Network Flow **9 Hour**

Greedy Algorithms – Huffman Codes, Activity Selection Problem, Amortized Analysis. Graph Algorithms – Topological Sorting, Minimum Spanning trees, Single Source Shortest Paths, Maximum Flow algorithms, Augmenting Paths, and Push-Relabel Methods. Network Flow – Ford-Fulkerson Algorithm, Push-relabel algorithm, Maximum bipartite matching, and Hopcroft-Karp algorithm. Parallel Algorithms – PRAM, Pointer Jumping and Parallel Prefix, Tree Contraction, Randomized Symmetry Breaking, Maximal Independent Set. Case Study: Finding the shortest path between two cities using Travelling salesperson problem

Unit-3 – Sorting Networks and Matrix Operations **9 Hour**

Sorting Networks: Comparison Networks, Zero-one principle, Bitonic Sorting Networks, Merging Network, Sorting Network. Matrix Operations- Strassen's Matrix Multiplication, inverting matrices, Solving system of linear Equations

Unit-4 – String Matching and Randomized Algorithms 9 Hour

String Matching –I String Matching, Rabin-Karp algorithm, Matching with finite Automata, Knuth-Morris-Pratt algorithm, Suffix Trees. Randomized Algorithms – Mathematical Background, Introduction and analysis, Monte Carlo Algorithms, Las Vegas Algorithms, Atlantic City Algorithms, Computational Complexity

Unit-5 – Approximation Algorithms and Linear Programming 9 Hour

Polynomial time, polynomial time verification, NP-Completeness and reducibility, NP-Complete problems. Approximation Algorithms- Vertex cover Problem, Travelling Sales person problem, Set Vector and Subset Sum. Geometric Algorithms – Preliminaries, Convex Hull, Segment Intersection, Closest Pair, Voronoi-Delaunay diagrams, Flip algorithm. Linear Programming – Framework, Formulation of Problems as Linear Programs. Duality. Simplex, Interior Point, and Ellipsoid Algorithms. Case Study : Types of algorithms used for finding convex hull for any given area.

Text Books / References

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, Third Edition, PHI.
2. Ellis Horowitz, Sartaj Sahni, and Rajasekharam, *Fundamentals of Computer Algorithms*, Galgotia Publications Pvt. Ltd.
3. Parag Himanshu Dave, Himanshu Bhalchandra Dave, *Design and Analysis Algorithms*, Pearson.
4. M. T. Goodrich and R. Tomassia, *Algorithm Design: Foundations, Analysis and Internet Examples*, John Wiley and Sons.
5. Allen Weiss, *Data Structures and Algorithm Analysis in C++*, Second Edition, Pearson Education.
6. Cormen, Leiserson, Rivest, and Stein, *Introduction to Algorithms*, Second Edition, MIT Press, Cambridge, MA, 2001. ISBN: 0262032937.

Advanced Computer Architecture

Course code: **L:T:P: 3:0:0**

Rationale:

The study of Advanced Computer Architecture equips students with the knowledge of modern computing systems and their performance-driven design principles. With the rapid evolution of processors, memory systems, and parallel architectures, understanding instruction-level parallelism, pipelining, multicore processing, and memory hierarchies has become essential for building efficient computing solutions. This course emphasizes quantitative design analysis, pipeline implementation, ILP exploitation, vector and SIMD architectures, GPU computing, and advanced memory optimizations. It enables students to critically analyze trade-offs in performance, energy, cost, and dependability, while preparing them for research and development in high-performance computing, embedded systems, and large-scale parallel architectures.

Course Outcomes:

CO1: Explain the principles of computer architecture, performance measurement techniques, and quantitative design trade-offs.

CO2: Analyze pipelining concepts, identify hazards, and evaluate pipeline implementation challenges using MIPS-based architectures.

CO3: Apply instruction-level parallelism techniques such as dynamic scheduling, branch prediction, and speculation to enhance processor performance.

CO4: Evaluate parallel architectures including SIMD, vector processors, GPUs, shared-memory multiprocessors, and multicore systems for computational efficiency.

CO5: Assess memory hierarchy designs, cache performance, and virtual memory optimizations to improve system throughput and reliability

Course Contents:

UNIT I

Introduction – Classes of computers – Defining Computer Architecture – Trends in Technology – Trends in Power and Energy in Integrated Circuits – Trends in Cost – Dependability – Measuring – Reporting and Summarizing Performance – Quantitative Principles of Computer Design.

UNIT II

Basic and Intermediate pipelining Concepts – The Major Hurdle of Pipelining – Pipeline Hazards – Pipelining Implementation – Implementation issues that makes Pipelining hard – Extending the MIPS Pipeline to Handle Multicycle Operations – The MIPS R4000 Pipeline**.**

UNIT III

Instruction-Level Parallelism: Concepts and Challenges – Basic Compiler Techniques for Exposing ILP – Reducing Branch Costs with Prediction – Overcoming Data Hazards with Dynamic Scheduling – Dynamic Scheduling – Hardware-Based Speculation – Exploiting ILP Using Multiple Issue and Static Scheduling – Exploiting ILP – Advanced Techniques for Instruction Delivery and Speculation – Studies of the Limitations of ILP.

UNIT IV

Vector Architecture – SIMD Instruction Set Extensions for Multimedia – Graphics Processing Units – Detecting and Enhancing Loop-Level Parallelism – Centralized Shared-Memory Architectures – Performance of Shared-Memory Multiprocessors – Distributed Shared Memory – Models of Memory Consistency – Multicore Processors and their Performance.

UNIT V

Review of Memory Hierarchy Design – Cache Performance – Basic Cache Optimizations – Virtual Memory – Protection and Examples of Virtual Memory – Advanced Optimizations of Cache Performance – Memory Technology and Optimizations – Protection: Virtual Memory and Virtual Machines – Crosscutting Issues: The Design of Memory Hierarchies – Case Studies / Lab Exercises.

Learning Resources:

Text Books

1. David. A. Patterson, John L. Hennessy, "Computer Architecture: A Quantitative approach", Fifth Edition, Elsevier, 2012.
2. K. Hwang, Naresh Jotwani, "Advanced Computer Architecture, Parallelism, Scalability, Programmability", Second Edition, Tata McGraw Hill, 2010.

Reference Books

1. V. Carl Hamacher, Zvonko G. Varanesic, Safat G. Zaky, "Computer Organisation", Sixth Edition, McGraw Hill Inc, 2012.
2. William Stallings, "Computer Organization and Architecture", Seventh Edition, Pearson Education, 2006.
3. Vincent P. Heuring, Harry F. Jordan, "Computer System Architecture", Second Edition, Pearson Education, 2005.

Artificial Intelligence

Course code: **L:T:P: 3:0:0**

Rationale:

Artificial Intelligence (AI) is one of the most transformative fields in modern computing, enabling machines to simulate intelligent human behavior. This course provides students with foundational knowledge of intelligent agents, problem-solving techniques, knowledge representation, planning, and reasoning under uncertainty. By integrating logic-based methods, probabilistic models, and real-world case studies, the course bridges theory with practical applications. Students will gain the skills necessary to analyze, design, and implement intelligent systems, preparing them for advanced research and careers in AI, machine learning, data science, robotics, and intelligent decision-making systems.

Course Outcomes:

CO1: Explain the fundamentals of artificial intelligence, characteristics of intelligent agents, and approaches to solving AI problems.

CO2: Apply various search strategies, including uninformed, informed, and heuristic-based methods, to solve optimization and decision-making problems.

CO3: Represent knowledge using first-order predicate logic and implement reasoning mechanisms such as forward chaining, backward chaining, and resolution.

CO4: Analyze planning techniques including state-space search, partial-order planning, and planning graphs for real-world applications.

CO5: Utilize probabilistic reasoning methods such as Bayesian networks and Hidden Markov Models to handle uncertainty in intelligent systems.

Course Contents:

UNIT I Introduction

Introduction–Definition– Future of Artificial Intelligence – Characteristics of Intelligent Agents – Typical Intelligent Agents – Problem Solving Approach to Typical AI problems.

UNIT II Problem Solving Methods

Problem solving Methods – Search Strategies – Uninformed – Informed – Heuristics – Local Search Algorithms and Optimization Problems – Searching with Partial Observations – Backtracking Search – Performance of search algorithms.

UNIT III Knowledge Representation

First Order Predicate Logic – Unification – Forward Chaining – Backward Chaining – Resolution – Knowledge Representation using First order Predicate logic – Reasoning Systems.

UNIT IV Planning

Planning with state-space search – partial-order planning – planning graphs – planning and acting in the real world – Plan generation systems.

UNIT V Uncertain Knowledge and Reasoning

Uncertainty – review of probability – probabilistic Reasoning – Bayesian networks – inferences in Bayesian networks – Temporal models – Hidden Markov models.

Learning Resources:

Text Book

1. S. Russel, P. Norvig, “Artificial Intelligence – A Modern Approach”, Third Edition, Pearson Education, 2015.

Reference Books

1. Kevin Night, Elaine Rich, Nair B., “Artificial Intelligence (SIE)”, Third Edition, McGraw Hill, 2017.
2. Dan W. Patterson, “Introduction to AI and ES”, Pearson Education, 2007.

Computational Number Theory

Course code: **L:T:P:**

Rationale:

Computational Number Theory combines the principles of number theory with algorithmic techniques to solve problems that are central to modern mathematics and cryptography. This course equips students with the mathematical foundations of integers, prime distribution, congruences, algebraic structures, and finite fields while emphasizing efficient algorithms for primality testing, factorization, discrete logarithms, and modular arithmetic. With a strong focus on applications such as cryptographic protocols (RSA, Diffie–Hellman), polynomial arithmetic, and computational techniques, the course prepares students for research and practice in theoretical computer science, cryptography, and advanced algorithm design.

Course Outcomes:

CO1: Apply fundamental concepts of integers, divisibility, congruences, arithmetic functions, and modular computation to solve computational number theory problems.

CO2: Analyze prime distribution theorems, implement probabilistic algorithms for prime generation, and apply cryptographic schemes such as RSA.

CO3: Demonstrate understanding of algebraic structures (groups, rings, fields, modules, vector spaces) and their computational applications.

CO4: Implement primality testing, factorization algorithms, discrete logarithm computation, and quadratic residuosity methods for cryptographic and number-theoretic problems.

CO5: Utilize advanced algorithms for discrete logarithms, polynomial arithmetic, linearly generated sequences, and finite field constructions in computational and cryptographic contexts.

Course Contents:

Unit–I

Basic properties of the integers – Divisibility and primality, Ideals and greatest common divisors, unique factorization, Congruences – Basic properties, Solving linear congruences, Residue classes, Euler's phi function, Fermat's little theorem, Arithmetic functions and Mobius inversion. Computing with large integers – Asymptotic notation, Machine models and complexity theory, Basic integer arithmetic, Computing in Zn, Faster integer arithmetic; Euclid's algorithm– basic Euclidean algorithm, extended Euclidean algorithm, computing modular inverses and Chinese remaindering, Speeding up algorithms via modular computation, Rational reconstruction and applications.

Unit–II

The distribution of primes – Chebyshev's theorem on the density of primes, Bertrand's postulate, Mertens' theorem, sieve of Eratosthenes, prime number theorem and beyond. Probabilistic algorithms – definitions, Approximation of functions, generating a random number from a given interval, generating a random prime, generating a random non-increasing sequence, generating a random factored number, RSA cryptosystem.

Unit–III

Algebraic Structures – Subgroups, Cosets and quotient groups, Group homomorphisms and isomorphisms, Cyclic groups, structure of finite abelian groups, Rings – Definitions, basic properties, and examples, Polynomial rings, Ideals and quotient rings, Ring homomorphisms and isomorphisms; Modules and vector

spaces – Submodules and quotient modules, Module homomorphisms and isomorphisms, Linear independence and bases, Vector spaces and dimension; Matrices – linear maps, inverse of a matrix, Gaussian elimination, Applications of Gaussian elimination; Algebras – The field of fractions of an integral domain, Unique factorization of polynomials, Polynomial congruences, Polynomial quotient algebras, General properties of extension fields, Formal power series and Laurent series, Unique factorization domains.

Unit–IV

Primality testing – Trial division, structure of Z_n, The Miller–Rabin test, Generating random primes using the Miller–Rabin test, Perfect power testing and prime power factoring, Factoring and computing Euler's phi function; Deterministic primality testing – The algorithm and its analysis.
Finding generators and discrete logarithms – Finding a generator for Z_p, Computing discrete logarithms Z_p, The Diffie–Hellman key establishment protocol.
Quadratic residues and quadratic reciprocity – Quadratic residues, Legendre symbol, Jacobi symbol; Computational problems related to quadratic residues – Computing the Jacobi symbol, Testing quadratic residuosity, Computing modular square roots, The quadratic residuosity assumption.

Unit–V

Subexpoential-time discrete logarithms and factoring – Smooth numbers, algorithm for discrete logarithms, algorithm for factoring integers, Practical improvements.
Polynomial arithmetic and applications – Basic arithmetic, Computing minimal polynomials, Euclid's algorithm, Computing modular inverses and Chinese remaindering, Rational function reconstruction and applications, Faster polynomial arithmetic; Linearly generated sequences and applications – Basic definitions and properties, Computing minimal polynomials, Solving sparse linear systems, The algebra of linear transformations.
Finite fields – The existence of finite fields, The subfield structure and uniqueness of finite fields, Conjugates, norms and traces; Algorithms for finite fields – Testing and constructing irreducible polynomials, Computing minimal polynomials in F[X]/(f), Factoring polynomials: the Cantor–Zassenhaus algorithm, Factoring polynomials: Berlekamp's algorithm, Deterministic factorization algorithms, Faster square-free decomposition.

Learning Resources:

Text Books

1. Victor Shoup, *A Computational Introduction to Number Theory and Algebra*, Cambridge University Press, 2008.
2. Henri Cohen, *A Course in Computational Algebraic Number Theory*, Springer-Verlag, 2000.

Reference Books

1. Abhijit Das, *Computational Number Theory*, Cambridge University Press, 2013.
2. Eric Bach and Jeffrey Shallit, *Algorithmic Number Theory, Volume 1: Efficient Algorithms*, MIT Press, 1996.
3. J. P. Buhler, P. Stevenhagen, *Algorithmic Number Theory: Lattices, Number Fields, Curves and Cryptography*, Cambridge University Press, 2008.

Programming Language Concepts

Course code: **L:T:P:**

Rationale:

Programming languages form the foundation of computer science, providing abstractions and paradigms to express algorithms and solve problems effectively. This course introduces students to the principles, design, and implementation of programming languages across different paradigms, including functional, logic, object-oriented, and parallel programming. It covers syntax, semantics, type systems, control mechanisms, and formal methods of program correctness. By studying multiple languages and their design philosophies, students gain the ability to evaluate trade-offs, write efficient and secure programs, and adapt to emerging languages and programming paradigms

Course Outcomes:

CO1: Explain the origins, abstractions, and paradigms of programming languages, and evaluate their design criteria such as efficiency, regularity, and security.

CO2: Apply functional and logic programming paradigms using languages like Scheme, ML, Haskell, and Prolog to solve computational problems.

CO3: Analyze object-oriented programming concepts, design issues, and implementation challenges across languages such as C++, Java, and Smalltalk.

CO4: Interpret and apply concepts of syntax, semantics, type systems, control structures, and abstract data types in programming language design.

CO5: Evaluate formal semantics, program correctness, and parallel programming constructs (threads, semaphores, monitors, message passing) in modern programming environments.

Course Contents:

Unit I

Introduction- The Origins of Programming Languages- Abstractions in Programming Languages – Computational Paradigms – Language Definition – Language Translation. Language Design Criteria – Efficiency, regularity, security and extensibility – C++: An Object-Oriented Extension of C – Python: A General-Purpose Scripting Language.

Unit II

Functional Programming – Programs as Functions – Scheme: A Dialect of Lisp – ML: Functional Programming with static typing – Delayed Evaluation – Haskell – Overloading.
Logic Programming – Logic and Logic Programs – Horn Clauses – Resolution and Unification – The Language Prolog – Problems with Logic Programming.

Unit III

Object-Oriented Programming – Software Reuse and Independence – Smalltalk – Java – C++ - Design Issues in Object-Oriented Languages – Implementation Issues in Object-Oriented Languages.
Syntax – Lexical Structure of Programming Languages – Context-Free Grammars and BNFs – Parse Trees and Abstract Syntax Trees – EBNFs and Syntax Diagrams – Parsing Techniques and Tools –Lexics vs. Syntax vs. Semantics.

Unit IV

Basic Semantics – Attributes, Binding, and Semantic Functions – Declarations, Blocks, and Scope – The Symbol Table – Name Resolution and Overloading – Allocation, Lifetimes, and the Environment – Variables and Constants – Aliases, Dangling References, and Garbage.

Unit V

Data Types – Data Types and Type Information – Simple Types – Type Constructors – Type Nomenclature in Sample Languages – Type Equivalence – Type Checking – Type Conversion – Polymorphic Type Checking – Explicit Polymorphism.

Control Expressions and Statements – Expressions – Conditional Statements and Guards – Exception Handling – Procedure Definition and Activation – Procedure Semantics – Parameter-Passing Mechanisms – Procedure Environments, Activations, and Allocation – Dynamic Memory Management – Exception Handling and Environments.

Unit VI

Abstract Data Types and Modules – The Algebraic Specification of Abstract Data Types – Abstract Data Type Mechanisms and Modules – Separate Compilation in C, C++ Namespaces, and Java Packages – Ada Packages – Modules in ML – Problems with Abstract Data Type Mechanisms.

Formal Semantics – A Sample Small Language – Operational Semantics – Denotational Semantics – Axiomatic Semantics – Proofs of Program Correctness.

Parallel Programming – Introduction to Parallel Processing – Parallel Processing and Programming Languages – Threads – Semaphores – Monitors – Message Passing.

Learning Resources:

Text Books

1. Kenneth C. Louden, *Programming Language Principles and Practices*, 2nd Edition, Thomson, 2003.
2. Carlo Ghezzi, Mehdi Jazayeri, *Programming Language Concepts*, 3rd Edition, John Wiley & Sons, 1997.

Reference Books

1. Robert W. Sebesta, *Concepts of Programming Languages*, 12th Edition, Pearson, 2019.
2. David A. Watt, *Programming Language Concepts and Paradigms*, Prentice-Hall, 1990.

Software Testing

Course code: **L:T:P:3:0:0**

Rationale:

Software testing plays a vital role in ensuring the quality, reliability, and correctness of software systems. It provides systematic techniques to detect and prevent defects throughout the software development life cycle. This course equips students with theoretical foundations and practical methodologies in testing strategies, test case design, testing levels, object-oriented testing, and monitoring/testing processes. By learning industry-standard approaches, students gain the skills to validate software functionality, performance, and security, thereby contributing to the development of robust and high-quality software.

Course Outcomes:

CO1: Explain the principles, processes, and maturity models of software testing and their role in software quality.

CO2: Apply black-box and white-box test case design strategies for defect detection and validation of software.

CO3: Analyze different levels of testing (unit, integration, system, and acceptance) and select appropriate strategies for various scenarios.

CO4: Implement testing techniques for object-oriented and real-time systems, including inter-class and class-level test design.

CO5: Evaluate and monitor testing activities using metrics, reviews, and test control methods for effective quality assurance.

Course Contents:

UNIT I Introduction

Software testing – The Role process in Software Quality – Testing as a process – Overview of testing maturity model – software testing definition – Software Testing Principles – Origin of defects – Defect classes – the defect Repository and Test Design.

UNIT II Test Case Design Strategies

Testing Strategies: Testing design strategies – Test case design strategies – Black box testing – Random Testing – Equivalence partitioning – Boundary value analysis – Cause-and-Effect – State transition – Error Guessing – COTS – White box testing techniques – Statement coverage – Branch Coverage – Condition coverage – Decision/Condition coverage – Multiple condition coverage – Dataflow coverage – Mutation testing.

UNIT III Levels of Testing

The Need for Levels of Testing – unit test – Planning – Designing the unit tests – Integration test – Integration Strategies for Procedure and Functions – Integration strategies for Classes – Integration test planning – System Test: Functional Testing – Performance Testing – Stress Testing – Configuration Testing – Security Testing – Recovery Testing – Regression testing – Alpha-Beta and Acceptance Tests.

UNIT IV Object Oriented Testing

Testing Object Oriented Software: Unit Testing in OO Context – Integration Testing in OO Context – OO testing methods – Class level testing – Interclass test case design – testing for real time system.

UNIT V Test Controlling and Monitoring

Controlling and Monitoring: Measurements and Milestone for Controlling and Monitoring: Status – Productivity – Cost – Error – fault and Failures – Effectiveness – Criteria for Test Completion – Reviews as testing Activity: Inspection Walkthrough – Components of review plan – testing for web application – Component level testing – clean room tests.

Learning Resources:

Text Books

1. Ilene Burnstein, “Practical Software Testing”, First Indian Reprint, Springer-Verlag, 2004.
2. S. Desai and A. Srivastava, “Software testing: A practical approach”, PHI Learning Pvt. Ltd., 2016.

Reference Books

1. Ali Behforooz, Frederick J Hudson, “Software Engineering Fundamentals”, Oxford University Press, New York, 2003.
2. William Perry, “Effective Methods for Software Testing”, John Wiley & Sons, Second Edition, USA, 2000.
3. Boris Beizer, “Software Testing Techniques”, Second Edition, Van Nostrand Reinhold, New York, 1990.
4. Aditya P. Mathur, “Foundations of Software Testing Fundamental Algorithms and Techniques”, Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008.
5. B. Homès, “Fundamentals of software testing”, John Wiley & Sons, 2024.

Computational Logic

Course code: **L:T:P: 3:0:0**

Rationale:

Computational logic provides the mathematical foundation for reasoning, problem-solving, and verification in computer science. This course introduces propositional, first-order, and modal logics, along with formal proof systems and resolution techniques. It equips students with the ability to model computational problems, reason about correctness, and apply logic to areas such as artificial intelligence, programming languages, and automated verification. Through the study of logical systems and their applications, students develop critical thinking and formal reasoning skills essential for advanced computing domains

Course Outcomes:

CO1: Demonstrate understanding of propositional logic, syntax, semantics, equivalences, and axiomatic systems.

CO2: Apply normal forms, resolution methods, and proof systems (e.g., natural deduction, sequent calculus, analytic tableaux) to logical problems.

CO3: Analyze first-order logic syntax, semantics, and metatheorems, and use axiomatic systems for reasoning and verification.

CO4: Evaluate and construct proofs in advanced logical frameworks including modal logic, natural deduction, and analytic tableaux.

CO5: Apply computational logic concepts to computer science applications such as satisfiability, formal verification, and computation tree logic.

Course Contents:

Unit-1 **9 Hour**

Propositional Logic-Introduction-Syntax of PL-Is It a Proposition? – Interpretations-Models-Interpretations-Equivalences and Consequences-More About Consequence-A Propositional Calculus-Axiomatic System PC-Five Theorems about PC-Using the Metatheorems-Adequacy of PC to PL-Compactness of PL

Unit-2 **9 Hour**

Normal Forms and Resolution-Truth Functions-CNF and DNF-Logic Gates-Satisfiability Problem-Resolution in PL-Resolution Strategies-Other Proof Systems for PL-Natural Deduction-Gentzen Sequent Calculus-Analytic Tableaux

Unit-3 **9 Hour**

First Order Logic-Syntax of FL-Scope and Binding-Substitutions-Semantics of FL-Translating into FL-Satisfiability and Validity-Some Metatheorems-A First Order Calculus-Axiomatic System FC-Six Theorems about FC-Adequacy of FC to FL-Compactness of FL

Unit-4 **9 Hour**

First Order Logic-Syntax of FL-Scope and Binding-Substitutions-Semantics of FL-Translating into FL-Satisfiability and Validity-Some Metatheorems-A First Order Calculus-Axiomatic System FC-Six Theorems about FC-Adequacy of FC to FL-Compactness of FL

Unit-5 **9 Hour**

Modal Logic K—Introduction-Syntax and Semantics of K-Validity and Consequence in K-Axiomatic System KC-Adequacy of KC to K-Natural Deduction in K-Analytic Tableau for K-Other Modal Logics-Various Modalities-Computation Tree Logic

Learning Resources:

Text Books:

1. Arindama Singh, *Logics for Computer Science*, PHI Learning Private Ltd, 2nd Edition, 2018.
2. Wasilewska & Anita, *Logics for Computer Science: Classical and Non-Classical*, Springer, 2018.

References:

1. Huth M. and Ryan M., *Logic in Computer Science: Modeling and Reasoning about Systems*, Cambridge University Press, 2005.
2. Dana Richards & Henry Hamburger, *Logic and Language Models for Computer Science*, Third Edition, World Scientific Publishing Co. Pte. Ltd, 2018.
3. Online Resource: Cornell University Lecture Notes on Logic

Neuro Fuzzy and Genetic Programming

Course code: **L:T:P: 3:0:0**

Rationale:

Soft Computing is an emerging field that integrates methodologies such as Neural Networks, Fuzzy Logic, Genetic Algorithms, and Hybrid Systems to solve complex, imprecise, and computationally intensive real-world problems. Unlike conventional computing approaches that rely on precise models, Soft Computing techniques are designed to handle uncertainty, approximation, and partial truth, making them highly effective in domains like optimization, pattern recognition, machine learning, control systems, and decision-making. This course provides students with both the theoretical foundation and practical skills required to design intelligent systems using computational models inspired by human reasoning and natural processes. Through hands-on practice with tools and case studies, students will gain the ability to model, analyze, and implement applications across engineering and scientific domains.

Course Outcomes:

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

CO1: Explain the fundamental concepts of Artificial Neural Networks, their architectures, learning algorithms, and applications.

CO2: Apply associative memory models and adaptive resonance theory (ART) to solve pattern recognition and classification problems.

CO3: Analyze fuzzy set theory, fuzzy logic systems, and defuzzification methods for handling uncertainty in real-world applications.

CO4: Implement and evaluate Genetic Algorithms for optimization problems using appropriate encoding, operators, and fitness functions.

CO5: Design and integrate hybrid soft computing systems combining Neural Networks, Fuzzy Logic, and Genetic Algorithms to address complex engineering problems.

.

Course Contents:

Unit-1 – Neural Networks **9 Hour**

Introduction to Artificial Intelligence Systems, Fundamentals of Neural Networks- Basic concepts, Human brain, Model of an Artificial Neuron, Neural Network Architectures, Characteristics of Neural Networks, Learning Methods, Taxonomy of Neural Network Architectures, History of Neural Network Research, Early Neural Network Architectures, McCulloch-Pitts Neuron Model; Back Propagation Networks – Architecture, Backpropagation Learning, Illustration, Applications, Effect of Turning Parameters of the Backpropagation Neural Network, Selection of various parameters in BPN, Variations of Standard Backpropagation Algorithms; Practice of Neural Network Tool – XOR Problem

Unit-2 – Associative Memory & Adaptive Resonance Theory 9 Hour

Associative Memory – Autocorrelators, Heterocorrelators, Wang et al.'s Multiple Training Encoding Strategy, Exponential BAM, Associative Memory for Real-coded Pattern Paris, Applications; Adaptive Resonance Theory – Introduction, ART1, ART2, Applications, Sensitivities of Ordering of Data; Practice of Neural Network Tool- Delta Rule

Unit-3 – Fuzzy Logic 9 Hour

Fuzzification; Fuzzy Set Theory- Fuzzy versus Crisp, Crisp Sets, Fuzzy Sets, Crisp Relations, Fuzzy Relations; Fuzzy Systems- Crisp Logic, Predicate Logic, Fuzzy Logic, Fuzzy Rule Based System, Defuzzification Methods, Practice of Fuzzy Logic Tool- Fuzzy Functions

Unit-4 – Genetic Algorithms 9 Hour

Genetic Algorithms- Basic Concepts, Creation of Offspring, Working Principle, Encoding, Fitness Function, Reproduction; Genetic Modelling – Inheritance Operators, Crossover, Inversion and Deletion, Mutation Operator, Bitwise Operators used in GA, Generation Cycle, Conversion of Genetic Algorithm, Applications, Multilevel Optimization, Advances in GA, Practice of Optimization in Genetic Algorithm Tool

Unit-5 – Hybrid Systems 9 Hour

Introduction, Neural Networks, Fuzzy Logic, and Genetic Algorithms Hybrids; Genetic Algorithm based Back Propagation Networks – GA Based Weight Determination; Fuzzy Backpropagation Networks- LR Type Fuzzy Numbers, Fuzzy Neuron, Fuzzy BP Architecture, Learning in Fuzzy BP; Fuzzy Logic Controlled Genetic Algorithms- GA in Fuzzy Logic Controller Design, Fuzzy Logic Controller, FLC-GA Based Structural Optimization.

Learning Resources:

Text Books:

1. S. Rajasekaran, G. A. Vijayalakshmi Pai, *Neural Networks, Fuzzy Logic, and Genetic Algorithms, Synthesis and Applications*, PHI Learning Private Limited, 15th Printing, 2011.
2. S. Rajasekaran, G. A. Vijayalakshmi Pai, *Neural Networks, Fuzzy Systems and Evolutionary Algorithms*, PHI Learning Private Limited, Second Edition, 2017.

Reference:

1. L. Fortuna, G. Rizotto, M. Lavorgna, G. Nunnari, M. G. Xibilia, and R. Caponetto, *Soft Computing, New Trends and Applications*, Springer, 2001.
2. S. N. Sivanandam, S. N. Deepa, *Principles of Soft Computing*, Wiley, 3rd Edition.
3. Dilip K. Prathihar, *Soft Computing – Fundamentals and Applications*, Alpha Science International Limited, 2014.
4. Simon Hawkins, *Neural Networks*, Pearson Education, 3rd Edition, 2008.
5. Y. A. Lone, H. Singh, *Deep Neuro-Fuzzy Systems with Python: With Case Studies and Applications from the Industry*, Apress, Germany, 2019.

Digital Image Processing

Course Contents:

UNIT I Introduction
Digital image representation, Fundamental steps in image processing, Components of Digital Image processing systems, Elements of visual perception, Image Formation model, Image Sampling and quantization, Relationship between pixels – neighborhood, adjacency connectivity, regions, boundaries and distance measures.

UNIT II Image Enhancement
Enhancement by point processing, Sample intensity transformation, Histogram processing, Image subtraction, Image averaging, Spatial filtering- Smoothing Spatial filters, Sharpening Spatial filters, Frequency domain- Fourier Transform, Low-Pass, High-Pass, Laplacian, Homomorphic filtering.

UNIT III Image Segmentation
Detection of discontinuities – point, line and edge detection, Edge linking and boundary detection, Thresholding, Region-based segmentation – region growing, region splitting and merging, Use of motion in segmentation- Spatial techniques and Frequency domain techniques.

UNIT IV Image Compression
Coding redundancy, Interpixel redundancy, fidelity criteria, Image compression models, Error-free compression, Variable length coding, Bit-plane coding, Lossless predictive coding, Lossy compression, Image compression standards, Real-Time image transmission JPEG and MPEG.

UNIT V Color Image Processing
Color Models, Pseudo color Image Processing, Color Transformations, Smoothing and sharpening, Image Segmentation based on color.

Learning Resources:

Text Books:

1. R.C. Gonzalez, R.E Woods, "Digital Image Processing", Pearson Education, 3rd Edition, 2008.

References:

1. R.C. Gonzalez, R.E Woods, S. L. Eddins, "Digital Image Processing Using MATLAB", PHI, 2003.
2. M. Sonka, V. Hlavac, R. Boyle, "Image Processing, Analysis, and Machine Vision", Brooks/Cole, 3rd Edition, 2007.
3. W.K. Pratt, "Digital Image Processing", Wiley-Interscience, 4th Edition, 2007.

Network Security

Course code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts of **information security** and its importance in modern computing.
2. To provide a comprehensive understanding of **cryptographic algorithms** and their application in securing data.
3. To enable students to identify various types of security threats and vulnerabilities in computer networks.
4. To equip students with the knowledge and skills to implement and manage **network security protocols** and security tools like firewalls and intrusion detection systems.

Course Outcomes:

CO1: To understand the basic concepts of network security, including security services, attacks, and mechanisms.
CO2: To apply various symmetric and asymmetric cryptographic techniques to ensure data confidentiality and integrity.
CO3: To implement cryptographic hash functions and digital signatures for authentication and message integrity.
CO4: To analyze and apply different network security protocols, such as SSL, TLS, and IPsec, to secure web and network communications.
CO5: To understand the role and functionality of **firewalls** and other security mechanisms in protecting networks from attacks.
CO6: To apply knowledge of cryptography and security principles to design and develop secure applications.

Course Contents:

UNIT I Overview of Network

Overview of Network Security – Security services – attacks – Security Issues in TCP/IP suite – Sniffing – spoofing – buffer overflow – ARP poisoning – ICMP Exploits – IP address spoofing – IP fragment attack – routing exploits – UDP exploits – TCP exploits.*

UNIT II Message Authentication Code

Authentication requirements – Authentication functions – Message Authentication Codes – Hash Functions – Security of Hash Functions and MACs – MD5 message Digest algorithm – Secure Hash Algorithm – RIPEMD – HMAC Digital Signatures – Authentication protocols – Kerberos – X.509.*

UNIT III IP Security

IP Security – AH and ESP – SSL/TLS – SSH – Web Security – HTTPS – DNS Security – Electronic Mail Security (PGP – S/MIME).*

UNIT IV Viruses

Intruders – Viruses – Worms – Trojan horses – Distributed Denial-Of-Service (DdoS) – Firewalls – IDS – Honey nets – Honey pots.*

UNIT V Introduction to Wireless Network Security

Introduction to wireless network security – Risks and Threats of Wireless networks – Wireless LAN Security (WEP – WPA).*

Learning Resources:

Text Books

1. W. Stallings, "Cryptography and Network Security: Principles and Practice", Fifth Edition, Prentice Hall, 2013.
2. Yang Xiao, Yi Pan, "Security in Distributed and Networking Systems", World Scientific, 2007.

References:

1. Aaron E. Earle, "Wireless Security Handbook", Auerbach Publications, Taylor & Francis Group, 2006.
2. Atul Kahate, "Cryptography and Network Security", Tata McGraw Hill, 2003.

PE-II (Any One From The Given Of PE-II)

Data Mining and Analytics

Course code: **L:T:P: 3:0:0**

Rationale:

1. To provide a foundational understanding of the core concepts and techniques of **data mining** and **data analytics**.
2. To equip students with the skills to preprocess raw data, extract meaningful patterns, and discover hidden insights from large datasets.
3. To introduce various algorithms and models for tasks such as **classification**, **clustering**, and **association rule mining**.
4. To enable students to apply data mining and analytics methods to solve real-world problems in various domains like business, science, and engineering.

Course Outcomes:

CO1: To understand the different phases of the Knowledge Discovery from Data (KDD) process and the functionalities of data mining.
CO2: To apply **data preprocessing** and **data cleaning** techniques to transform raw data into a suitable format for analysis.
CO3: To implement and evaluate different **classification models** like decision trees and support vector machines.
CO4: To analyze and apply **clustering algorithms** to group similar data instances and discover patterns.
CO5: To discover frequent itemsets and derive **association rules** from transactional data.
CO6: To comprehend and apply techniques for mining and analyzing complex data types, such as text, web, and time-series data.

Course Contents:

Unit-1 – Data Mining Introduction (9 Hours)
Introduction: Kinds of Data- Kinds of Patterns-Data Objects and Attribute Type- Data Visualization -Data Preprocessing: Data cleaning, Data Integration, Data Transformation, Data Discretization and Data Reduction: Attribute Subset Selection-Histograms, Clustering, Sampling

Unit-2 – Associations and Correlations (9 Hours)
Market Basket Analysis – Apriori Algorithm – Mining Frequent Itemsets without Candidate Generation – Mining Frequent Itemsets Using Vertical Data Format – Mining Closed Frequent Itemsets – Mining Multilevel Association Rules – Mining Multidimensional Association Rules – Correlation Analysis – Constraint-Based Association Mining

Unit-3 – Classification and Prediction (9 Hours)
Basic Concepts- Decision Tree Induction-Attribute selection Measures-ID3 and CART algorithms, Tree Pruning-Bayes Classification Methods: Bayes‟ Theorem, I Bayesian Classification – Classification by Backpropagation- Support Vector Machines-Lazy learners: KNN-Metrics for evaluating classifier performance-Techniques to improve classification accuracy-Prediction: Regression Analysis

Unit-4 – Cluster Analysis (9 Hours)

Cluster Analysis: Partitioning Methods- Hierarchical Methods: Agglomerative versus Divisive Hierarchical Clustering-Probabilistic Model based Clustering – BIRCH, DBSCAN, STING, CLIQUE Techniques-Evaluation of clustering Techniques

Unit-5 – Outliers and Statistical Approaches in Data Mining (9 Hours)

Introduction to outliers, Challenges in detecting Outliers, Outlier Detection Methods – Supervised, Semisupervised, Unsupervised- Statistical Data Mining approaches – Data mining in Recommender Systems, Data mining for Intrusion Detection, Data Mining for Financial Analysis

Learning Resources:

Text Books

1. Jiawei Han and Micheline Kamber, *Data Mining Concepts and Techniques*, Third Edition, Elsevier, 2012.
2. Ian H. Witten, Eibe Frank and Mark A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, Fourth Edition, Elsevier, 2017.

Reference Books

1. Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, *Introduction to Data Mining*, Addison Wesley, 2005.
2. Margaret H. Dunham, *Data Mining: Introductory and Advanced Topics*, Pearson Education, 2006.
3. Arun K. Pujari, *Data Mining Techniques*, Universities Press, 2013.
4. Charu C. Aggarwal, *Data Mining: The Textbook*, Springer, 2015.
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Second Edition, Springer, 2009.

Distributed Operating Systems

Course code: **L:T:P:3:0:0**

Rationale:

1. To introduce the fundamental concepts, architectures, and challenges of **distributed operating systems (DOS)**.
2. To explore the key design issues in building a DOS, such as **concurrency, synchronization**, and **fault tolerance**.
3. To provide a detailed understanding of how resources are managed in a distributed environment, including **distributed file systems** and **memory management**.
4. To enable students to compare and contrast the different models and technologies used for communication and coordination in a distributed system.

Course Outcomes:

CO1: To understand the principles and characteristics of distributed systems and their architectural models.
CO2: To analyze and apply various **inter-process communication (IPC)** mechanisms and remote procedure calls (RPCs) in a distributed environment.
CO3: To comprehend the challenges of **distributed synchronization** and apply solutions for mutual exclusion and deadlock handling.
CO4: To evaluate different **distributed file system (DFS)** models and their performance and consistency issues.
CO5: To understand the concepts of **distributed shared memory** and its implementation challenges.
CO6: To analyze and apply **fault tolerance** and **recovery techniques** to build robust distributed systems.

Course Contents:

Unit-1 – Introduction to Distributed System **9 Hour**

Distributed Systems- Goals of Distributed Systems, Hardware Concepts- Bus-based Multiprocessors- Switched Multiprocessors, - Bus-based Multicomputers- Switched Multicomputers, Software Concepts- Network Operating System – True Distributed Systems- Multiprocessors- Time Sharing Systems- Design Issues-Distributed Systems. Case Study: Mach OS System Architecture

Unit-2 – Fundamentals of Communication systems **9 Hour**

Layered Protocols, ATM networks, Client Server model – Blocking Primitives, Non-Blocking Primitives, Buffered Primitives, Unbuffered Primitives, Reliable primitives, Unreliable primitives, Message passing and its related issues, Remote Procedure Call and its related issues, Case Studies: Communication in Mach OS.

Unit-3 – Synchronization in Distributed Systems **9 Hour**

Fundamentals of Clock Synchronization, Logical clock, Physical clock, Algorithms for Clock synchronization, Mutual Exclusion-Centralized Algorithm, Distributed Algorithm, Token Ring Algorithm, Comparison of all three algorithms, Importance of Election Algorithm, Bully Algorithm, Ring Algorithm, Atomic Transaction- Introduction Transaction Model, Concurrency Control, Deadlock in Distributed Systems, Distributed Deadlock Detection. Synchronization in Mach OS.

Unit-4 – Processes and Processors in Distributed Operating Systems **9 Hour**

Threads, Design issues of Threads package, Workstation Model, System Model – Introduction, Using Idle Workstations, Processor Pool Model, Hybrid Model, Processor Allocation – Allocation Model, Design issues for processor Allocation Algorithms, Example of processor Allocation Algorithms, Scheduling in Distributed Systems, Load Balancing and Sharing Approach, Fault Tolerance- Fault tolerance Using Active Replication, Primary-backup, Real Time Distributed Systems- Communication, Real Time Scheduling. Case study – Process Management in Mach OS

Unit-5 – Distributed Shared memory **9 Hour**

Introduction, Bus-Based Multiprocessors, Switched Multiprocessors, Ring-based Multiprocessors, Numa Multiprocessors, Consistency Models – Strict Consistency, Casual Consistency, PRAM Consistency, Weak Consistency, Release Consistency, Entry Consistency, Page Based Distributed Shared Memory – Replication, granularity, Finding the Owner, Finding the Copies, Page Replacement, Synchronization, Shared – Variable Distributed Shared memory, Object Based Distributed Shared memory, Case Study – Memory Management in Mach OS

Learning Resources:

Text Books:

1. Andrew S. Tanenbaum, *Distributed Operating Systems*, Pearson Education, 2011.
2. Pradeep K. Sinha, *Distributed Operating Systems Concepts and Design*, PHI, 2012.
3. Mukesh Singhal, Niranjan G. Shivratri, *Advanced Concepts in Operating Systems*, McGraw Hill International, 2017.

References:

1. Kayhan Erciyes, *Distributed Real-Time Systems: Theory and Practice*, Springer, 2019.
2. http://www.cs.iit.edu/~sun/cs550.html

Pattern Recognition Techniques

Course code: **L:T:P: 3:0:0**

Rationale:

- To introduce the fundamental concepts, principles, and methodologies of **pattern recognition**.
- To provide a comprehensive understanding of various techniques for **feature extraction, classification**, and **clustering**.
- To explore the mathematical and statistical foundations underlying pattern recognition algorithms.
- To equip students with the skills to design and implement pattern recognition systems for real-world applications.

Course Outcomes:

CO1: To understand the basic concepts of pattern recognition and its relationship with machine learning and data mining.
CO2: To apply different techniques for **preprocessing** and **feature extraction** from raw data.
CO3: To implement and evaluate various **classification algorithms**, such as **Bayesian classifiers**, **k-Nearest Neighbors**, and **Support Vector Machines**.
CO4: To analyze and apply **unsupervised learning techniques**, including clustering algorithms like **k-means** and **hierarchical clustering**.
CO5: To evaluate the performance of pattern recognition systems using appropriate metrics.
CO6: To apply pattern recognition techniques to solve real-world problems in areas like image processing, speech recognition, and bioinformatics.

Course Contents:

Unit-1 – Introduction to Pattern Recognition Systems **9 Hour**
Basics of Probability, Independence of events, Conditional and Joint probability, Machine perception, Pattern Recognition Systems, The Design cycle, Learning and Adaptation, Minimum-error-rate classification, Classifiers, Discriminant functions, Decision surfaces, Normal density, and discriminant functions

Unit-2 – Parameter Estimation Methods **9 Hour**
Maximum Likelihood Estimation, Bayesian Estimation, Bayesian Parameter Estimation: Gaussian case, Bayesian Parameter Estimation: General Theory, Problems of Dimensionality, Component Analysis and Discriminants, Expectation-Maximization, Hidden Markov Model

Unit-3 – Non-Parametric Techniques **9 Hour**
Density Estimation, Parzen Windows, K- Nearest Neighbor Estimation, The Nearest Neighbor Rule, Metrics and Nearest Neighbor classification, Fuzzy classification, reduced coulomb energy networks, Approximations by series expansions

Unit-4 – Linear Discriminant Functions **9 Hour**
Linear Discriminant Functions and Decision Surfaces, Generalized Linear Discriminant Functions, Two-category linearly separable case, Perceptron criterion functions, Relaxation procedures, Nonseparable Behavior, Minimum Squared Error procedures, Linear Programming Algorithms, Support Vector Machines

Unit-5 – Multilayer Neural Networks and Nonmetric Methods 9 Hour

Introduction to Neural Networks. Multilayer Neural Networks: Feedforward operations and classification, Backpropagation algorithms, Nonmetric methods: Decision Trees, CART, Applications: Face recognition System

Learning Resources:

Text Books:

1. S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4th Edition, Academic Press, 2009

References:

2. C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006
3. P. A. Devijver and J. Kittler, *Pattern Recognition: A Statistical Approach*, Prentice-Hall International, Englewood Cliffs, NJ, 1980
4. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd Edition, Academic Press, New York, 1990
5. Jianxin Wu, *Essentials of Pattern Recognition: An Accessible Approach*, Cambridge University Press, United Kingdom, 2020
6. Y. Bengio, I. Goodfellow, A. Courville, *Deep Learning*, MIT Press, United Kingdom, 2016

Biometrics

Course code: **L:T:P:3:0:0**

Rationale:

1. To introduce the fundamental concepts and principles of **biometric recognition systems**.
2. To provide a comprehensive understanding of various biometric modalities, including physiological and behavioral traits.
3. To explore the key components and processes involved in a biometric system, such as **feature extraction** and **matching**.
4. To discuss the security, privacy, and ethical issues associated with the use of biometrics.

Course Outcomes:

CO1: To understand the basic principles and applications of biometric systems.
CO2: To analyze and implement different **physiological biometric modalities**, such as fingerprint, face, and iris recognition.
CO3: To evaluate various **behavioral biometric modalities**, including signature, voice, and gait recognition.
CO4: To comprehend the different phases of a biometric system, from **data acquisition** to **decision making**.
CO5: To analyze the performance of a biometric system using metrics like False Acceptance Rate (FAR) and False Rejection Rate (FRR).
CO6: To discuss the security vulnerabilities, privacy concerns, and societal implications of using biometrics.

Course Contents:

Unit-1 – Introduction **9 Hour**

Basics of biometric systems, Biometric functionalities: verification, identification- Introduction to unimodal system, Introduction to multimodal system, what is image, acquisition, type, point operations, Geometric transformations-First and Second Derivatives- steps in edge detection, smoothening, enhancement, thresholding, localization, Low level feature extraction, Describing image motion- High level feature extraction, Template matching

Unit-2 – Process of Biometric System **9 Hour**

Biometrics Sensors, Data Acquisition and Database, Biometrics Pre-processing Techniques-Image restoration and segmentation, Pattern Extraction and Classification, Fingerprint Identification Technology-Fingerprint Patterns, Fingerprint Features, Fingerprint Image, width between two ridges -Fingerprint Image Processing – Minutiae Determination – Fingerprint Matching: Fingerprint Classification, Matching policies

Unit-3 – Multibiometric System **9 Hour**

Introduction to Multibiometric – Information Fusion in Biometrics – Issues in Designing a Multibiometric System – Sources of Multiple Evidence – Levels of Fusion in Biometrics – Sensor level, Feature level, Rank level, Decision level fusion – Score level Fusion. Introduction to various matching methods – LDA, PCA, Eigen Vectors and Values-Covariance, Correlation- Introduction to decision theory and their examples

Unit-4 – Authentication Procedure **9 Hour**

physiological and behavioral properties of biometric system, Software biometrics systems, Hardware biometrics systems, Security of biometric systems- Advisory, insider, infrastructure attacks- Attacks at the

user interface- impersonation, obfuscation, spoofing Attacks on system module and interconnections- Counter measure: Biometric template security- Challenges in biometric systems like fool proofing, false positives

Unit-5 – Applications **9 Hour**

access control like a lock or an airport check-in area- immigration and naturalization- welfare distribution- military application- banking, e.g., check cashing, credit card, ATM- computer login; intruder detection; smart card- multi-media Communication; WWW and an electronic purse- sensor fusion; decision fusion- categorization: e.g., age and gender- industrial automation – efficient enrollment gesture interpretation; on-line shopping- other commercialized service: Fingerprint, Face detection, Irish Recognition

Learning Resources:

Text Books:

1. James Wayman, Anil Jain, Davide Maltoni, Dario Maio, *Biometric Systems: Technology Design and Performance Evaluation*, Springer, 2005.
2. James Wayman, Anil K. Jain, Arun A. Ross, Karthik Nandakumar, *Introduction to Biometrics*, Springer, 2011.
3. Mark S. Nixon, Alberto S. Aguado, *Feature Extraction and Image Processing for Computer Vision*, Third Edition, Elsevier, 2012.

References:

1. Rafael C. Gonzalez, Richard Eugene Woods, *Digital Image Processing using MATLAB*, Second Edition, Tata McGraw-Hill Education, 2019.
2. Ruud M. Bolle, Sharath Pankanti, Nalini K. Ratha, Andrew W. Senior, Jonathan H. Connell, *Guide to Biometrics*, Springer, 2009.
3. Richard O. Duda, David G. Stork, Peter E. Hart, *Pattern Classification*, Wiley, 2007.

Shimon K. Modi, *Biometrics in Identity Management: Concepts to Applications*, Artech House, 2011.

Information Storage and Management

Course code: **L:T:P:3:0:0**

Rationale:

1. To understand the fundamental principles and technologies of information storage in modern data center environments.
2. To provide knowledge about various storage architectures, including Direct-Attached Storage (DAS), Storage Area Networks (SAN), and Network-Attached Storage (NAS).
3. To explore the concepts of storage virtualization, cloud storage, and data protection strategies like backup and replication.
4. To enable students to comprehend the challenges in managing vast amounts of information and to learn the tools and techniques for effective storage management

Course Outcomes:

CO1: To understand the evolution of storage technology and the key components of a data center infrastructure.
CO2: To describe and analyze different storage networking technologies, including FC SAN, IP SAN, and NAS.
CO3: To explain the purpose and implementation of RAID technology for data protection.
CO4: To comprehend the concepts of business continuity and various data replication and backup strategies.
CO5: To identify and evaluate various forms of storage virtualization and their benefits.
CO6: To apply knowledge of storage security and management to ensure data availability, integrity, and confidentiality.

Course Contents:

Unit-1 – Introduction to Information Storage and Management **9 Hour**
Introduction to Information Storage Management- Evolution of Storage Architecture- Data Centre Infrastructure- Evaluate storage architectures and key data center elements in classic, virtualized and cloud environments- physical and logical components of a storage infrastructure including storage subsystems, RAID and intelligent storage systems- Intelligent Storage Array T1: A hospital application stores the Patient records with core elements of the data center. Discussion of typical challenges the storage management team may face in meeting the service-level demands of the hospital staff
T2: Design RAID- Real Life Scenario Business Application for Acme Telecom

Unit-2 – Introduction to SAN, IPSAN and CAS **9 Hour**
storage networking technologies -SAN- FCSAN- IP-SAN, IPSAN-iSCSI components FcoE- NAS and object-based, and unified storage (CAS) – NAS I/O Operations- NAS Implementations
T5:USRobotics Mini NAS provides easy backup and remote access for college office setting
T6: SAN Infrastructure implementation at Heterogeneous environment
T7: Configuration and Tracing of FC scan and ISCSI scan

Unit-3 – Introduction to Electronic Storage Information **9 Hour**
Electronically Stored Information and the Federal Rules of Civil Procedure- Changes to the Federal Rules of Civil Procedure-Federal Rules of Evidence-The (Long) List of Stakeholders-Ownership of Data-Data Control Considerations-Business Continuity (BC) Terminology, BC Planning life cycle- Failure Analysis, Business Impact Analysis-Automatic Path fail over-Backup granularities

T7: Discussion of Social Media Cases and Spoliation Cases
T8: Use Backup techniques for archived data
T9: Implement appropriate backup a"dre'tore model in virtual environments for sensitive information

Unit-4 – Introduction to Storage Security, Recovery and Acquisition 9 Hour

Applications as a Vital User Interface-Hidden or Restricted Access Data -Privileged, Sensitive, and Inaccessible Data Management-Proving Ownership and Integrity-Legal and Forensically Sound Acquisition – Securing the Data-Access Control and Management-Organization and File Management Techniques-Safe Storage Issues and Considerations Information Security Framework-Risk Triad-Storage Security Domains.
T10: Analyzevarious monitoring parameters in secured storage environment
T11: Creation of linux Instance in Public Cloud Generate a private key, access using SSH client
T12: Usage of RSAandVmware Security Products

Unit-5 – Cloud Storage 9 Hour

key characteristics, services, deployment models, and infrastructure components for a cloud computing-Cloud Infrastructure Mechanism-Cloud Infrastructure Mechanism: Cloud Infrastructure Mechanism-Logical Network Perimeter-Cloud Usage Monitor-Cloud Adoption Considerations-Cloud Storage Gateways
T13: Building programs to deploy cloud publications
T14: Usage of Cloud services with open-source cloud tools (like Eucalyptus, Openstack, Open Nebula and others)

Learning Resources:

Text Books:

1. David R. Matthews, *Electronically Stored Information: The Complete Guide to Management, Understanding, Acquisition, Storage, Search, and Retrieval*, Second Edition.
2. EMC Corporation, *Information Storage and Management*, 2nd Edition, Wiley India, ISBN: 978-1118094839.

References:

1. Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, 2013, ISBN: 9780133387568.
2. UifTroppen, Rainer, Wolfgang Muller, *Storage Networks Explained*, Wiley, India, 2010, ISBN: 978-0470741436.

Wireless Sensor Networks

Course code: **L:T:P:**

Rationale:

1. To introduce the fundamental concepts and principles of wireless sensor networks (WSNs) and their unique characteristics.
2. To explore the various layers of the WSN protocol stack and the challenges in designing protocols for resource-constrained sensor nodes.
3. To provide a comprehensive understanding of localization, time synchronization, and data aggregation techniques in a WSN.
4. To enable students to design and implement sensor network applications for different domains like environmental monitoring and smart homes.

Course Outcomes:

CO1: To understand the architecture and components of a wireless sensor node and the different types of WSNs.
CO2: To analyze and evaluate different MAC protocols specifically designed for WSNs.
CO3: To apply routing protocols for efficient data delivery and energy conservation in a sensor network.
CO4: To comprehend the challenges and solutions for localization and time synchronization in WSNs.
CO5: To understand and implement data aggregation and dissemination techniques to reduce network traffic

Course Contents:

Unit-1 – Introduction to Wireless Sensor Networks **9 Hour**
Motivation for a network of Wireless Sensor nodes – Sensing and sensors- Challenges and constraints- Node architecture- Sensing sub system- Processor sub system- Communication interfaces–prototypes- Application of Wireless sensors- WSN Tools- Overview and Limitations- Contiki -Introduction- Characteristics of Contiki WSN simulator- Sensor Network Programming- Challenges in Sensor Network Programming- Node-Centric Programming

Unit-2 – MAC Protocols for WSN **9 Hour**
Characteristics of MAC protocols in Sensor networks- Contention free MAC Protocols- MAC Protocols – Characteristics- Traffic Adaptive Medium Access- Y-MAC- Low energy Adaptive Clustering- Contention based MAC Protocols- Sensor MAC- Timeout MAC and pattern MAC- Hybrid MAC Protocol- MAC protocols in ContikiOS simulator- Nullmac in Contiki simulator

Unit-3 – Routing Strategies **9 Hour**
Routing Metrics- Flooding and Gossiping- Data-Centric Routing- Proactive Routing- On-Demand Routing- Hierarchical Routing- Location-Based Routing- QoS-Based Routing Protocols- Local Power Management Aspects- Dynamic Power Management

Unit-4 – Localization and Time Synchronization **9 Hour**

Clocks and the Synchronization Problem- Time Synchronization in Wireless Sensor Networks- Basics of Time Synchronization- Time Synchronization Protocols- Ranging Techniques- Range-Based Localization- Range-Free Localization- Event-Driven Localization

Unit-5 – WSN Security and Middleware Services **9 Hour**

Fundamentals of Network Security- Challenges of Security in Wireless Sensor Networks- Security Attacks in Sensor Networks- Protocols and Mechanisms for Security- Middleware Architecture- Data related functions, Architecture- Existing middleware- MiLAN, IrisNet- AMF, DSWare- CLMF

Learning Resources:

Textbooks

1. Waltenegus Dargie, Christian Poellabauer, *Fundamentals of Wireless Sensor Networks, Theory and Practice*, Wiley Series on Wireless Communication and Mobile Computing, 2011.
2. Kazem Sohraby, Daniel Minoli, *Wireless Sensor Networks – Technology, Protocols and Applications*, Wiley Interscience Publications, 2013.
3. S. Swapna Kumar, *A Guide to Wireless Sensor Networks*, Kindle Edition, USP Publications, 2017.

References

1. C.S. Raghavendra, Krishna M. Sivalingam, Taieb Znati, *Wireless Sensor Networks*, Springer Science, 2010.
2. Bhaskar Krishnamachari, *Networking Wireless Sensors*, Cambridge University Press, 2005.
3. A Guide to Wireless Sensor Networks – Kindle Edition (Amazon)
4. Contiki Tutorials – USC ANRG
5. Waltenegus Dargie, *Fundamentals of Wireless Sensor Networks* [PDF Download] (file link)

Network Protocols and Programming

Course code: **L:T:P: 3:0:0**

Rationale:

1. To understand the fundamental principles of computer networks and the role of various protocols in enabling communication between devices.

2. To provide a comprehensive understanding of the TCP/IP protocol suite and its layers.

3. To acquire practical skills in network programming, specifically using socket programming to develop networked applications.

4. To enable students to analyze and troubleshoot network traffic and performance issues

Course Outcomes:

CO1: To understand the basic concepts of networking, protocol layering, and different network architectures like OSI and TCP/IP.
CO2: To analyze and implement various protocols at the application layer, such as HTTP, DNS, and FTP.
CO3: To comprehend the functions of the transport layer, including TCP and UDP, and apply them in programming.
CO4: To understand the principles of IP addressing, routing, and the functionalities of the network layer.
CO5: To develop client-server applications using socket programming in a language like Python or C++.
CO6: To analyze and troubleshoot network communication issues using tools like Wireshark.

Course Contents:

Unit-1 – Network and Transport Layer Protocols **9 Hour**
OSI Model, IP Header, ARP, RARP, ICMP, UDP Data gram and its characteristics, TCP Header & Connection Establishment Process, IP Fragmentation, ARP, RARP, ICMP, UDP Data gram and its characteristics, TCP Header & Connection Establishment Process

Unit-2 – Socket Programming **9 Hour**
Byte Ordering Conversion Functions, man page, System Calls used with sockets, Iterative and Concurrent Server, Socket Interface, Remote Procedure Call, TCP Client Server Program, UDP Client Server Program

Unit-3 – Application Layer Protocols **9 Hour**
DNS, TELNET, FTP, HTTP, WWW, DHCP, SMTP, POP3 Multicasting Protocols BGMP, IGMP, MBGP, MOSPF, PGM, PIM-DM PIM-SM

Unit-4 – Ipv6 Protocol **9 Hour**
IPV6 Features, Ipv6 Header, Ipv6 Address Types, Ipv6 Routing Protocols, Ipv4 to Ipv6 Tunneling and Translation Techniques
Unit-5 – WAN Protocols **9 Hour**
DSL and Cable technology, Frame Relay, ATM, PPP, HDLC, MPLS, CR-LDP, LDP, RSVP-TE

Learning Resources:

Textbooks

1. Forouzan, Behrouz A. *TCP/IP Protocol Suite*. McGraw-Hill Higher Education, 2002.
2. Stevens, W. Richard, Andrew M. Rudoff, and Bill Fenner. *Unix Network Programming Volume 1: The Sockets Networking API*. Vol. 3. Boston: Addison-Wesley Professional, 2003.

3. Dong, Jielin. *Network Protocol Handbook*. 2007.

References

1. Beej's Guide to Network Programming Using Internet Sockets. Brian "Beej Jorgensen" Hall, v3.1.9, Copyright © November 16, 2022.
2. https://people.scs.carleton.ca/~lanthier/teaching/COMP1406/Notes/COMP1406_Ch12_NetworkProgramming.pdf
3. https://www.geeksforgeeks.org/types-of-network-protocols-and-their-uses/
4. https://internalpointers.com/post/network-programming-beginners-overview

Network Routing Algorithms

Course code: **L:T:P:3:0:0**

Rationale:

1. To understand the fundamental principles and concepts of network routing, which is critical for the efficient and reliable delivery of data packets.

2. To explore and analyze various routing algorithms, including both static and dynamic approaches, and their suitability for different network environments.

3. To comprehend the design and functionality of key routing protocols used in the Internet, such as RIP, OSPF, and BGP.

4. To gain the skills to evaluate the performance of different routing algorithms based on metrics like latency, throughput, and scalability.

Course Outcomes:

CO1: To understand the role and importance of routing in computer networks and the factors that influence routing decisions.
CO2: To analyze and compare the performance of distance vector and link-state routing algorithms.
CO3: To comprehend the design and implementation of hierarchical routing and its application in large-scale networks.
CO4: To evaluate and apply congestion control algorithms to manage network traffic and improve performance.
CO5: To understand the principles of broadcast and multicast routing and their applications.
CO6: To apply knowledge of routing algorithms to design and troubleshoot routing in real-world network scenarios

Course Contents:

Unit-1 – Routing Foundations **9 Hour**
Introduction to Networking and Network Routing – IPV4 Addressing, Protocol Stack Architecture – Router Architecture – Network Topology Architecture – Network Management Architecture – Global Telephone Network – Communication Technologies – Standards Committees.

Unit-2 – Routing Algorithms: Shortest Path, Widest Path, and Spanning Tree **9 Hour**
Bellman–Ford Algorithm: Centralized View – A Distance Vector Approach: Distributed View – Dijkstra's Algorithm: Centralized Approach – Dijkstra's Algorithm: Distributed Approach – Widest Path Algorithm: Dijkstra-Based Approach – Widest Path Algorithm: Distance Vector-Based Approach – Spanning Tree: Breadth First Search and Depth First Search – Minimum Spanning Tree – Steiner Tree Algorithms – k-Shortest Paths Algorithm

Unit-3 – Routing Protocols: Framework and Principles **9 Hour**
Routing Protocol, Routing Algorithm, and Routing Table – Routing Information Representation and Protocol Messages – Distance Vector Routing Protocol: Illustration – Distance Vector Routing Protocol: Times – Babel Routing Protocol – Link State Protocol: In-Band Hop-by-Hop Dissemination – Link State Protocol: In-Band Based on End-to-End Session – Path Vector Routing Protocol – Link Cost

Unit-4 – IP Routing and Distance Vector Protocol Family **9 Hour**
Routers, Networks, and Routing Information – Routing Table – Communication of Routing Information – Static Routes – Routing Information Protocol, Version 1 (RIPv1) : Communication and Message Format –

RIPv1 General Operation – Routing Information Protocol, Version 2 (RIPv2) – Interior Gateway Routing Protocol (IGRP) – Enhanced Interior Gateway Routing Protocol (EIGRP)

Unit-5 – Router Architecture and Design **9 Hour**

Router Architectures – Functions of a Router – Types of Routers – Elements of a Router – Packet Flow – Packet Processing: Fast Path Versus Slow Path – Router Architectures, Impact of Addressing on Lookup – Longest Prefix Matching

Learning Resources:

Textbooks

1. Deepankar Medhi, Karthik Ramasamy, *Network Routing Algorithms, Protocols, and Architectures*, Morgan Kaufmann Publishers, Second Edition, 2017.
2. James Aweya, *IP Routing Protocols*, CRC Press, First Edition, 2021.

References

1. Yoram Orzach, Deepanshu Khanna, *Network Protocols for Security Professionals*, Packt Publishers, First Edition, 2022.
2. James H. Baxter, *Wireshark Essentials*, Packt Publishers, First Edition, 2017.

High Performance Computing

Course code: **L:T:P-3:0:0**

Rationale:

1. To introduce the fundamental concepts and principles of high-performance computing (HPC) and its importance in modern scientific and engineering domains.
2. To provide a comprehensive understanding of various parallel and distributed computing architectures, including multicore, GPU, and cluster computing.
3. To explore parallel programming models and tools for developing efficient and scalable applications.
4. To enable students to analyze the performance of parallel programs and optimize them for speed and efficiency.

Course Outcomes:

CO1: To understand the basic concepts of parallel computing, including different architectures and parallel programming models.
CO2: To analyze and evaluate the performance of parallel algorithms and identify factors that affect performance.
CO3: To apply shared-memory programming models like OpenMP to develop parallel applications.
CO4: To implement distributed-memory programming using Message Passing Interface (MPI) for cluster computing.
CO5: To comprehend the principles of GPU computing and apply frameworks like CUDA or OpenCL for accelerating applications.
CO6: To apply knowledge of HPC to solve complex problems in various scientific and data-intensive domains.

Course Contents:

Unit-1 – Fundamentals of Parallel Processors **9 Hour**

Stored Program Computer Architecture- General purpose cache- based microprocessor-Performance based metrics and benchmarks- Moore's Law- Pipelining- Superscalarity- SIMD- Memory Hierarchies Cachemapping- prefetch- Multicore processors- Mutithreaded processors- Vector Processors- Design Principles- Maximum performance estimates- Programming for vector architecture – Data flow computers and VLSI Computations – Need for Migrating to Nanoscale Processors – Design of Quantum Processors and Quantum Logic gates with Qubits Processing

Unit-2 – Performance Enhancement Computing Cluster Computing and Sky Computing 9 Hour

Introduction to Cluster Computing- Scalable Parallel Computer Architectures- Cluster Computer and its Architecture- Classifications, Components for Clusters- Cluster Middleware and Single System Image- Resource Management and Scheduling, Programming Environments and Tools, Applications, Representative Cluster Systems, Heterogeneous Clusters, Security, Resource Sharing, Locality, Dependability, Cluster Architectures, Detecting and Masking Faults, Recovering from Faults, Condor, Evolution of Metacomputing. Virtualised Architecture for Cloud Computing Storage, Hypervisor usage and Integration of Cloud Computing for Sky Computing Model

Unit-3 – Perspective of Parallel Algorithms **9 Hour**

Principles of parallel algorithm design – Data Parllel, Task graph, Work pool, master- slave, pipeline, Hybrid – Non –numerical algorithms, sorting, graph algorithms, search algorithms for discrete optimization problems,Dynamic programming – Numerical algorithms, Dense matrix algorithms, fast Fourier transforms

Unit-4 – Constructs of Parallel Programming 9 Hour

Introduction to parallel computing – parallel programming platforms – Basic communication operations – Programming using message passing paradigm, MPI– Programming shared address space platforms, POSIX threads, open MP

Unit-5 –Gpu Architecture and Programming 9 Hour

Hardware Architecture – Integrated GPUs –Multi GPUs – GPU Architecture – Memory Handling with CUDA: Shared Memory, Global Memory, Constant Memory and Texture Memory.Introduction to CUDA C, parallel programming in CUDA C , Thread cooperation- Shared Memory and Synchronization

Learning Resources:

Text Books:

1. Georg Hager, Gerhard Wellein, *Introduction to High Performance Computing for Scientists and Engineers*, Chapman & Hall / CRC Computational Science series, 2011.
2. Parag K. Lala, *Quantum Computing: A Beginners Introduction*, McGraw Hill, 2020.
3. R. Buyya, *High Performance Cluster Computing: Architectures and Systems, Volume 1*, Pearson Education, 2008.
4. Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar, *Introduction to Parallel Computing*, 2nd edition, Addison-Wesley, 2003.
5. Nicholas Wilt, *CUDA Handbook: A Comprehensive Guide to GPU Programming*, Addison-Wesley, 2013.
6. Jason Sanders, Edward Kandrot, *CUDA by Example: An Introduction to General Purpose GPU Programming*, Addison-Wesley, 2010.

References:

1. https://link.springer.com/article/10.1007/s10586-017-0727-5
2. http://www.nvidia.com/object/cuda_home_new.html

Cyber Security Essentials for AI & IoT

Course code: **L:T:P-3:0:0**

Rationale:

In today's hyper-connected world, cyber security has become a fundamental requirement for safeguarding personal data, organizational assets, digital infrastructure, and national security. The exponential growth of digital transactions, social media platforms, and connected devices has widened the threat landscape, leading to new forms of cybercrime, data breaches, and security risks. This course on Cyber Security Essentials is designed to build a strong foundation in understanding cyber threats, laws, data privacy, digital payments, and emerging technologies. Through a balanced mix of conceptual knowledge, case studies, and hands-on exposure, the course prepares students to critically analyze security challenges, adopt preventive measures, and implement best practices in securing cyberspace.

Course Outcomes (COs)

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

CO1: Explain the fundamentals of cyberspace, cyber threats, and the importance of cyber security in modern society.

CO2: Identify and analyze different types of cybercrimes, understand relevant cyber laws, and evaluate ethical and legal implications.

CO3: Apply principles of data privacy and data security to real-world scenarios, including challenges in social media and big data environments.

CO4: Assess security issues in e-commerce and digital payment systems, and recommend secure practices for digital transactions.

CO5: Evaluate emerging cyber security challenges (IoT, cloud, mobile), and demonstrate awareness of monitoring tools, best practices, and real-world case studies.

Course content:

Unit 1: Foundations of Cyber Security

Defining cyberspace; overview of computer & web technologies; internet & web infrastructure; architecture of cyberspace; governance & regulation; concepts, issues & challenges in cyber security. Threat landscape: types of threats, cyber-attack vectors, basic defensive strategies.

Unit 2: Cyber Crime, Cyber Law, & Ethical Issues

Classification of cyber crime: computer & mobile devices, social engineering, financial fraud, crimes against children & women. Malware, ransomware, zero-day & zero-click attacks; modus operandi of cyber criminals; contract & law in India: IT Act and amendments; international perspectives. Ethical issues, privacy, responsible disclosure, digital rights.

Unit 3: Data Privacy, Data Security, & Social Media Challenges

Definitions: data, metadata, personal vs non-personal data; principles of data protection; big data security issues. Relevant privacy laws: GDPR, Indian Personal Data Protection Bill, PIPEDA etc. Social media: privacy risks, content moderation, challenges in monitoring & security, case studies of data breaches.

Unit 4: E-Commerce, Digital Payments & Security Practices

Components & structure of e-commerce systems; major threats & best practices (secure payments, secure transactions, authentication, encryption). Digital payment modes: cards, UPI, wallets etc.; stakeholders; risk & trust models. Security in web applications and APIs; SSL/TLS basics; secure coding practices; incident response.

Unit 5: Emerging Topics, Monitoring & Case Studies *(or Tools & Practical)*

Emerging threats: IoT security, cloud security, mobile security challenges. Security best practices and standards; monitoring & auditing; security policies; vulnerability assessment & penetration testing overview. Real-world case studies of cyber crimes, data breaches, lessons learned. Optional lab / practical session: Configure simple security tools; simulate attacks & defenses; privacy settings; basic incident handling.

Learning Resources

Textbooks

1. **William Stallings, Lawrie Brown**, *Computer Security: Principles and Practice*, Pearson, 4th Edition, 2018.
2. **Mark Ciampa**, *Security+ Guide to Network Security Fundamentals*, Cengage Learning, 7th Edition, 2022.
3. **B. B. Gupta, D. P. Agrawal**, *Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, Wiley, 2014.

Reference Books

1. **Charles P. Pfleeger, Shari Lawrence Pfleeger, Jonathan Margulies**, *Security in Computing*, Pearson, 5th Edition, 2015.
2. **Michael T. Simpson, Kent Backman, James Corley**, *Hands-On Ethical Hacking and Network Defense*, Cengage Learning, 2019.
3. **Matt Bishop**, *Computer Security: Art and Science*, Addison-Wesley, 2nd Edition, 2018.
4. **Ross Anderson**, *Security Engineering: A Guide to Building Dependable Distributed Systems*, Wiley, 3rd Edition, 2020.
5. **P. W. Singer, Allan Friedman**, *Cybersecurity and Cyberwar: What Everyone Needs to Know*, Oxford University Press, 2014.

Entrepreneurship Management

Course code: **L:T:P:**

Rationale:

1. To understand the fundamental concepts of entrepreneurship, including the qualities of an entrepreneur and the process of identifying business opportunities.

2. To explore the different stages of a startup lifecycle, from idea generation and market research to business plan development and funding.

3. To acquire knowledge of the legal, financial, and marketing aspects of starting and managing a new venture.

4. To enable students to develop a comprehensive business plan and pitch it effectively to potential investors.

Course Outcomes:

CO1: To comprehend the core concepts of entrepreneurship and the role of innovation in creating new ventures.
CO2: To identify and evaluate business opportunities by conducting thorough market research and feasibility analysis.
CO3: To formulate a detailed business plan that includes financial projections, marketing strategies, and operational plans.
CO4: To understand the different sources of funding for startups, such as venture capital, angel investors, and crowdfunding.
CO5: To analyze the legal and ethical considerations involved in starting and running a business.
CO6: To develop presentation and communication skills to effectively pitch a business idea to stakeholders.

Course Contents:

UNIT – I Introduction to Entrepreneurship:
Definition of Entrepreneur Entrepreneurial Traits. Entrepreneur vs Manager, creating and starting the venture: sources of new ideas, method of generating ideas, creative problem solving – writing business plan, evaluating business plans. Launching formalities.

UNIT – II Financing and Managing the new ventures:
sources of capital, record keeping, recruitment, motivating and leading teams, financial controls. Marketing and sales controls. E commerce and Entrepreneurship, internet advertising – new venture expansion strategies and issues.

UNIT – III Industrial Financial Support:
schemes and functions of directorate of industries, District industries centre (DICs) Industrial development corporation (IDC), State Financial corporation (SFCs), small scale industries development corporation (SSIDCs) Khadhi and village industries commission (KVIC) Technical Consultancy 57arbonizatio (TCO), Small industries service institute (SISI), national small industries corporation (NSIC), small industries development bank of india (SIDBI).

UNIT – IV Production and marketing management:
Thrust areas of production management, selection of production techniques, plant 57arbonizati and maintenance, designing the work place, inventory control, material handling and quality control. Marketing functions, market segmentation market research and channels of distribution, sales promotion and product pricing.

UNIT – V

Labour legislation, salient provision of health, safety, and welfare under Indian factories Act, Industrial dispute act, employees state insurance act, workmen's compensation act and payment of bonus act.

Learning Resources:

Text Books:

1. Dollinger: Entrepreneurship, Pearson, 2009.
2. Robert Hisrich, & Michael Peters: Entrepreneurship, TMH, 2009.

References:

1. Vasant Desai, Dynamics of Entrepreneurial Development and Management, Himalaya Publishing House, 2009.
2. Harvard Business Review on Entrepreneurship, HBR Paper Back.
3. Robert J. Calvin: Entrepreneurial Management, TMH, 2009.
4. Gurmeet Naroola: The Entrepreneurial Connection, TMH, 2009.
5. Bolton &Thompson: Entrepreneurs—Talent, Temperament and Techniques, Butterworth Heinemann, 2009.
6. Agarwal: Indian Economy, Wishwa Prakashan, 2009.
7. Dutt & Sundaram: Indian Economy, S. Chand, 2009.
8. B D Singh: Industrial Relations &Labour Laws, Excel, 2009.
9. Aruna Kaulgud: Entrepreneurship Management, Vikas Publishing House, 2009.
10. Thomas W. Zimmerer & Norman M. Scarborough: Essentials of Entrepreneurship and Small Business Management, PHI, 2009.
11. N.D. Kapoor: Industrial Law, Sultan Chand & Sons, 2009.

Practical

Business Communications

Course code: **L:T:P: 0:0:2**

Rationale:

1. To equip engineering students with the essential communication skills required for professional success in the corporate world.

2. To provide a foundational understanding of various forms of business communication, both written and verbal, and their specific contexts.

3. To develop proficiency in creating professional documents such as notices, memos, reports, and proposals.

4. To enhance interpersonal and soft skills critical for effective team collaboration, presentations, and cross-cultural interactions.

Course Outcomes:

CO1: To demonstrate a strong command of business vocabulary and apply it in drafting formal written communication.
CO2: To effectively draft and format a variety of business documents, including notices, memos, letters, and reports.
CO3: To improve speaking skills for professional scenarios, such as conducting meetings and delivering presentations.
CO4: To apply active listening and verbal communication skills in different business contexts, including cross-cultural settings.
CO5: To understand and practice the principles of group dynamics and negotiation for effective team collaboration.
CO6: To comprehend and implement appropriate communication strategies for different professional situations.

Course Contents:

Unit 1
Business Vocabulary – Writing: Drafting Notices, Agenda, and Minutes – Reading: Business news, Business articles.

Unit 2
Writing: Style and vocabulary – Business Memorandum, letters, Press Releases, reports – proposals – Speaking: Conversational practice, telephonic conversations, addressing a gathering, conducting meetings.

Unit 3
Active Listening: Pronunciation – information gathering and reporting – Speaking: Cross-Cultural Issues, Group Dynamics, negotiation & persuasion techniques.

Learning Resources:

Text Books

1. Jones, Leo & Richard Alexander. New International Business English. CUP. 2003.
2. Horner, David & Peter Strutt. Words at Work. CUP. 1996.

3. Levi, Daniel. Group Dynamics for Teams. 3rd ed. Sage Publications India Pvt. Ltd., New Delhi, 2011.
4. Owen, Roger. BBC Business English. BBC. 1996.
5. Henderson, Greta Lafollette & Price R Voiles. Business English Essentials. 7th Edition. Glencoe / McGraw Hill.
6. Sweeney, Simon. Communicating in Business. CUP. 2000.

Lab-I
(Computer Network LAB)

Course code: **L:T:P:0:0:2**

Rationale:

1. To provide a hands-on experience in understanding the practical aspects of computer networks and their protocols.

2. To acquire skills in configuring network devices, analyzing network traffic, and implementing network security measures.

3. To enable students to apply theoretical knowledge of networking concepts to solve real-world problems.

4. To develop proficiency in using network simulation tools and network analysis software.

Course Outcomes:

CO1: To understand and implement network topology configurations, including star, bus, and mesh, using network simulation tools.
CO2: To analyze the functions of different layers of the OSI and TCP/IP models using packet sniffers like Wireshark.
CO3: To apply subnetting and supernetting concepts to design and manage IP addressing schemes.
CO4: To implement client-server communication using socket programming and various transport layer protocols.
CO5: To configure and troubleshoot network devices like routers and switches to enable data communication.
CO6: To implement and evaluate basic network security measures such as firewalls and access control lists

List of experiments:

1. Simulation experiments for Network and Transport layer protocols performance – OSPF, RIP, BGP, TCP, IP, etc.
2. Configuring, testing and measuring network devices (e.g., switches, routers, firewalls) and parameters/policies (interfaces, bandwidth, access control, etc.)
3. Network management experiments
4. Exercises on network programming
5. Projects on Network Service Management and Communications

Learning Resources:

Text Books

1. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, *Compilers: Principles, Techniques and Tools*, Addison-Wesley.
2. Michael L. Scott, *Programming Language Pragmatics*, Elsevier.

Reference Books

1. Andrew W. Appel, *Modern Compiler Implementation in C/Java*, Cambridge University Press.
2. Keith D. Cooper and Linda Torczon, *Engineering a Compiler*, Elsevier.
3. Allen I. Holob, *Compiler Design in C*, Prentice-Hall.
4. Steven S. Muchnik, *Advanced Compiler Design and Implementation*, Elsevier.
5. Randy Allen and Ken Kennedy, *Optimizing Compilers for Modern Architectures*, Elsevier.

Lab-II
(Software Engineering LAB)

Course code: **L:T:P:**

Rationale:

1. To provide a practical understanding of the principles and methodologies of **software engineering**.
2. To enable students to apply different **software development lifecycle models** to a project.
3. To acquire hands-on experience in using **software engineering tools** for requirements analysis, design, and testing.
4. To develop proficiency in working in a team environment to build a quality software product.

Course Outcomes:

CO1: To apply different **software process models** to a given software development project.
CO2: To prepare **Software Requirement Specification (SRS)** documents using various techniques.
CO3: To design software systems using **Unified Modeling Language (UML)** diagrams, such as use case, class, and sequence diagrams.
CO4: To apply different **testing techniques** to ensure the quality and reliability of the software.
CO5: To use various software engineering tools to manage different phases of the software development life cycle.
CO6: To comprehend and implement **project management** concepts, including cost estimation and risk analysis.

List of experiments:

1. Introduction and project definition
2. Introduction to Project Management Software
3. Software process overview
4. Project planning
5. Software Requirements Specification (SRS)
6. Introduction to UML and use case diagrams
7. Flow of events and activity diagram
8. Object-Oriented analysis: discovering classes and class diagrams
9. Interaction diagrams: sequence and collaboration diagrams
10. Software Design: software architecture and object-oriented design
11. Effort and cost estimation using COCOMO
12. State Transition Diagram
13. Component and deployment diagrams
14. Issue Tracking Systems like GitHub, BitBucket
15. Software Configuration Management tools like CVS or SVN
16. Software testing

Jharkhand University of Technology, Ranchi

B.Tech
Computer Science and Engineering

NEP-2020 based Syllabus w.e.f – 2025-26 Batch

Semester- VIth

(Compiler Design)

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To understand the fundamental principles, design, and implementation of compilers, which are essential for translating high-level programming languages into machine code.

2. To provide a comprehensive understanding of the different phases of a compiler, including lexical analysis, parsing, semantic analysis, and code generation.

3. To explore the various algorithms and data structures used in each phase of the compilation process.

4. To enable students to design and implement a simple compiler or components of a compiler

Course Outcomes:

CO1: To comprehend the overall structure and working of a compiler and its different phases.
CO2: To implement a lexical analyzer to identify tokens and use tools like Flex/Lex.
CO3: To apply different parsing techniques, such as top-down and bottom-up parsing, for syntax analysis.
CO4: To understand and implement intermediate code generation and apply various optimization techniques.
CO5: To analyze the principles of runtime environments and the challenges of code generation.
CO6: To apply knowledge of compiler design to develop a simple programming language or a component of a compiler.

Course Contents:

Unit-1 – Introduction **12 Hour**

Compilers-Phases of Compiler-Cousins of the Compiler-Grouping of Phases-Compiler construction tools-Lexical Analysis-Role of Lexical Analyzer-Input Buffering -Specification of Tokens -LEX -Finite Automata-Regular Expressions to Automata -Minimizing DFA

Unit-2 – Top Down Parsing **12 Hour**

Role of Parser-Grammars-Error Handling-Context-Free Grammars-Writing a grammar- Elimination of Ambiguity-Left Recursion- Left Factoring-Top Down Parsing — Recursive Descent Parser- Predictive Parser- LL(1) Parser- Computation of FIRST-Computation of FOLLOW-Construction of a predictive parsing table-Predictive Parsers LL(1) Grammars- Predictive Parsing Algorithm- Problems related to Predictive Parser – Error Recovery in Predictive Parsing

Unit-3 – Bottom-Up Parsing **12 Hour**

Bottom Up Parsing-Reductions-Handle Pruning-Shift Reduce Parser-Problems related to Shift Reduce Parsing-Operator Precedence Parser, LEADING, TRAILING -LR Parser- LR Parsers- Need of LR Parsers-LR (0)Item-Closure of Item Sets- Construction of SLR Parsing Table -Problems related to SLR-Construction of Canonical LR(1)- Problems related to CLR – LALR Parser— Problems related to LALR-YACC

Unit-4 – Code Generation **12 Hour**

Intermediate Code Generation- prefix – postfix notation- Quadruple – triple – indirect triples Representation-Syntax tree- Evaluation of expression – Three-address code- Synthesized attributes – Inherited attributes – Intermediate languages – Declarations- Assignment Statements- Boolean Expressions- Case Statements- Back patching – Procedure calls- Code Generation- Issues in the design of code generator- The target machine –

Runtime Storage management- A simple Code generator- Code Generation Algorithm- Register and Address Descriptors

Unit-5 – Code Optimization **12 Hour**

Code optimization -Principal Sources of Optimization- Function Preserving Transformation- Loop Optimization- Peephole optimization — DAG- Basic Blocks- Flow Graphs- Global Data Flow Analysis — Efficient Data Flow Algorithm- Runtime Environments- Source Language issues- Storage Organization- Activation Records- Storage Allocation strategies

Learning Resources:

Textbooks:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, *Compilers: Principles, Techniques and Tools,* Second Edition, Pearson Education, 2011.
2. S. Godfrey Winster, S. Aruna Devi, R. Sujatha, *Compiler Design*, Yesdee Publishing Pvt. Ltd., 2016.
3. K. Muneeswaran, *Compiler Design*, Oxford Higher Education, Fourth Edition, 2015.

References:

David Galles, *Modern Compiler Design*, Pearson Education, Reprint 2012.

1. Raghavan V., *Principles of Compiler Design*, Tata McGraw Hill Education Pvt. Ltd., 2010.

AI and Machine Learning

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and principles of Artificial Intelligence (AI) and its subfield, Machine Learning (ML).

2. To provide a comprehensive understanding of various AI and ML algorithms, including supervised, unsupervised, and reinforcement learning.

3. To explore the process of building, training, and evaluating machine learning models for solving real-world problems.

4. To equip students with the skills to apply AI and ML techniques in various domains like computer vision, natural language processing, and data analytics.

Course Outcomes:

CO1: To understand the basic concepts, history, and applications of Artificial Intelligence and its key areas.
CO2: To apply supervised learning algorithms for solving classification and regression problems.
CO3: To implement unsupervised learning techniques for clustering and dimensionality reduction.
CO4: To analyze and evaluate the performance of different AI and ML models using appropriate metrics.
CO5: To comprehend the principles of reinforcement learning and apply them to simple problems.
CO6: To apply knowledge of AI and ML to develop intelligent systems for real-world applications

Course Contents:

UNIT I
AI – History of AI – Agents – Structure of Intelligent Agents – Environments – Problem Solving Methods – Problem Solving Agents – Formulating Problems – Search Strategies – Breadth-First – Uniform Cost – Depth-First – Depth-Limited – Bidirectional– Informed Search – Best-First Heuristic Functions – Memory Bounded Search – A* – SMA* – Iterative Improvement Algorithms – Hill Climbing – Simulated Annealing – Measure of Performance and Analysis of Search Algorithms.

UNIT II
Game Playing – Perfect Decisions – Imperfect Decisions – Alpha-Beta Pruning – Knowledge Based Agent – Wumpus World Environment – Propositional Logic – Agent for Wumpus World – First Order Logic – Syntax – Semantics – Extensions – Using First Order Logic – Representation Change in the World – Goal Based Agents.

UNIT III
Knowledge Base – Knowledge Representation – Production Based System – Frame Based System – Inference – Backward Chaining – Forward Chaining.

UNIT IV
Learning from Agents – Inductive Learning – Types of Machine Learning – Supervised Learning – Learning Decision Trees – Support Vector Machines – Neural and Belief Networks – Perceptron – Multi-layer Feed Forward Networks – Bayesian Belief Networks.

UNIT V

Unsupervised Learning – K-means Clustering – Hierarchical Clustering – Agglomerative and Divisive Clustering – Fuzzy Clustering.

Learning Resources:

Text Books

1. Stuart Russel, Peter Norvig, "AI – A Modern Approach", Second Edition, Pearson Education, 2007.
2. Kevin Night, Elaine Rich, Nair B., "Artificial Intelligence (SIE)", McGraw Hill, 2008.

Reference Books

1. Vinod Chandra SS, Anand Hareendran S, "Artificial and Machine Learning", First Edition, PHI Learning, 2014.
2. Dan W. Patterson, "Introduction to AI and ES", Pearson Education, 2007.
3. G. Luger, W. A. Sttubblefield, "Artificial Intelligence", Third Edition, Addison-Wesley Longman, 1998.
4. N. J. Nilson, "Principles of Artificial Intelligence", Narosa Publishing House, 1980.
5. Tom Mitchell, "Machine Learning", First Edition, Tata McGraw Hill India, 2017.

PROGRAMME ELECTIVES III & IV

PE-III (Any One FromThe Given of PE-III)

Database Security and Privacy

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To understand the fundamental concepts of database security and the importance of protecting data in a modern, data-driven world.

2. To explore the various threats and vulnerabilities to databases and the mechanisms to counter them.

3. To provide a comprehensive understanding of access control models, authentication methods, and cryptographic techniques used for securing data.

4. To discuss the legal and ethical issues related to data privacy and compliance regulations like GDPR

Course Outcomes:

CO1: To understand the basic security goals for a database system, including confidentiality, integrity, and availability.
CO2: To apply various access control models to enforce security policies and restrict unauthorized access to data.
CO3: To analyze and implement authentication and authorization mechanisms for securing database transactions.
CO4: To comprehend the principles of data encryption and apply it to secure sensitive information in a database.
CO5: To understand the challenges of data privacy and implement techniques for data anonymization and privacy-preserving data mining.

Course Contents:

Unit-1-Security Architecture **9 Hour**

Introduction, Information Systems, Database Management Systems, Information Security Architecture, Database Security, Asset Types and Their value, Security Methods. Operating System Security Fundamentals: Introduction, Operating System Overview, The Components of an Operating System Security Environment, Authentication Methods, User Administration, Password Policies, Vulnerabilities of Operating Systems, Email Security, Internet security

Unit-2 - Administration of Users **9 Hour**

Introduction, Documentation of User Administration, Operating System Authentication, Creating Users, creating a SQL Server user, removing users, Modifying users, Default users, Remote users, Database Links, Linked Servers, Remote Servers Practices for Administrators and Managers, Profiles, Password Policies. Privileges and Roles: Introduction, Defining and Using Profiles, Designing and Implementing Password Policies, Granting and Revoking User Privileges, Creating, Assigning and Revoking User Roles, Best practices

Unit-3 - Database Application Security Models **9 Hour**

Types of Users, Security Models, Application Types, Application Security Models, Transparent Data Encryption, Column Encryption Full Database Encryption, Column level Security with SQL Server. Virtual Private Databases: Implementation of VPD using Views, Application Context in Oracle, Implementing Oracle VPD, Viewing VPD Policies, VPD using views, Application contexts using Data Dictionary, Policy Manager Implementing Row and Column level Security with SQL Server

Unit-4 - Audits **9 Hour**

Ways to Audit a Database, Application API Code, Auditing with Trigger Code, Normal Audit, Unified Audit, Fine Grained Auditing, Comparing Methods Auditing Database Activities: Oracle Database Activities, Creating DLL Triggers with Oracle, Auditing Database Activities with Oracle, Auditing Server Activity with SQL Server 2022, Auditing Server Activity with Oracle21c, Case study: project security and auditing

Unit-5 - Privacy Preserving Techniques **9 Hour**

Privacy Preserving Data Mining Techniques, Privacy Preserving Data Mining Models and Algorithms-The Randomization Method, Group Based Anonymization, Distributed Privacy-Preserving Data Mining, Privacy-Preservation of Application Results, and Applications of Privacy-Preserving Data Mining. Case study: Joseph Domingo-Ferrer, Nine Measures of Anonymity, k-Anonymous Data Mining

Learning Resources:

Text Books

1. Hassan A. Afyouni, "D“tabase Security and Auditing Protecting Data Integrity and Accessibility",”Third Edition, Cengage Learning, 2013.
2. Ron Ben Natan, "I“plementing Database Security and Auditing",”Elsevier Digital Press, 2005.

Reference Books

1. Osama Mustafa and Robert P. Lockard, "O“acle Database Application Security with Oracle Internet Directory, Oracle Access Manager, and Oracle Identity Manager",”First Edition, Apress Publishers, 2019.
2. Charu C. Aggarwal and Philip S. Yu, "P“ivacy Preserving Data Mining Models and Algorithms",”Kluwer Academic Publishers, 2008.

Software Defined Networks

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To understand the fundamental principles and architecture of Software Defined Networking (SDN), which decouples the control and data planes of a network.
2. To explore the key components of an SDN architecture, including the controller, data plane devices, and Southbound and Northbound APIs.
3. To acquire knowledge of various SDN controllers and programming paradigms for building network applications.
4. To enable students to analyze the benefits of SDN in terms of network agility, flexibility, and management, and its applications in cloud computing and data centers

Course Outcomes:

CO1: To comprehend the evolution of networking and the core concepts and architecture of SDN.
CO2: To analyze the functions and components of the SDN control plane and different types of controllers.
CO3: To implement network functionalities using SDN programming and various APIs like OpenFlow.
CO4: To understand the principles of Network Function Virtualization (NFV) and its relationship with SDN.
CO5: To evaluate the security challenges and solutions in a software-defined network.

Course Contents:

Unit- - Introduction to SDN **(9 Hours)**
Introduction – Evolution of SDN – Need for SDN – Generalized and Distributed Control and Data Planes – The Genesis of SDN – Introducing Mininet – Setting up the Environment and Implementation of SDN Aspects in Mininet

Unit-2 - SDN Abstractions **(9 Hours)**
Working Principle of SDN – OpenFlow Protocol – SDN Controllers: Introduction – General Concepts – VMware – Circa – OpenFlow-Related – Mininet – NOX/POX – Trema – Ryu – Floodlight – Layer 3 Centric – Plexxi – Cisco OnePK – Implementation of Custom Topologies in POX, ODL, Floodlight, Click, ONOS – Interfacing: Northbound – Southbound – Eastbound – Westbound

Unit-3 - Programming of SDN **(9 Hours)**
Network Programmability – Network Function Virtualization – NetApp Development – Network Slicing – ONOS Deployment – ONOS – OPNFV

Unit-4 - SDN Applications and Use Cases **(9 Hours)**
SDN in the Data Center – SDN in Other Environments – SDN Applications – SDN Use Cases – The Open Network Operating System

Unit-5 - SDN's Future and Perspectives **(9 Hours)**
SDN Open Source – SDN Futures – SDN Security – Switching and Load Balancers – Firewall and Access Control – Use Cases in Legacy Networks – Security

Learning Resources:

Textbooks

1. Software Defined Networks: A Comprehensive Approach by Paul Goransson and Chuck Black, Morgan Kaufmann Publications, 2016 Reprint
2. SDN – Software Defined Networks by Thomas D. Nadeau & Ken Gray, O'Reilly, 2013
3. Software Defined Networking with OpenFlow By Siamak Azodolmolky, Packt Publishing, 2013

Reference

4. 4. Feamster, Nick, Jennifer Rexford, and Ellen Zegura. "The road to SDN: an intellectual history of programmable networks." ACM SIGCOMM Computer Communication Review 44.2 (2014): 87-98

Semantic Web

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and vision of the Semantic Web, which aims to make internet data machine-readable.

2. To provide a comprehensive understanding of the key technologies and standards that form the foundation of the Semantic Web, such as RDF, RDFS, and OWL.

3. To explore the principles of knowledge representation and ontology engineering for building intelligent and interoperable systems.

4. To enable students to design and query semantic data and apply these techniques to solve real-world problems.

Course Outcomes:

CO1: To understand the basic concepts of the Semantic Web and its difference from the traditional web.
CO2: To comprehend and apply Resource Description Framework (RDF) for representing data on the web.
CO3: To implement and query semantic data using languages like SPARQL.
CO4: To analyze and apply Web Ontology Language (OWL)for building and reasoning with ontologies.
CO5: To understand the principles of Linked Data and its role in connecting disparate datasets.

Course Contents:

Unit-1 –Introduction: Web Intelligence **9 Hour**

Web Information System Environment and Foundations: Web information description and query languages-the semantic Web-Web information syst
em development tools, Web Human-Media Engineering: Web page design-information representation-information processing-visualization of Web information, Web Information Management: information transformation, Internet and Web-based data management: OLAP (online analytical processing)- Web knowledge management- web page automatic generation and updating-Web security, integrity, privacy and trust. The Web Operation- Parsing Common Document Types: Representing Styled Text- Approaches for Modeling the Web User Behaviour
T1: Ruby's TextResource class: Plain Text, Binary Document Formats, HTML and XHTML, OpenDocument, RSS
T2: HandlinI Document Formats (PDF)
T3: HandlinI Document Formats (Word)

Unit-2 - Web Usage Data Pre-Processing **9 Hour**

Data Sources and Collection: Nature of Web Usage Data- Merging and Cleaning Data, Web Session Reconstruction and User Identification: Heuristics and Methods for Sessionization- Dealing with Incomplete Information- Model Assessment for Sessionization Algorithms. Discovering Usage Patterns for Web: Web Usage Data Collection- Web Usage Data Preprocessing- Web Usage Pattern Discovery - Application of Web Usage Patterns.
Cleaning, Segmenting, and Spell-Checking Text: Extracting All Text from Any XML File- Segmenting Text-Stemming Text.

T4: Spell-Checking Text
T5: Recognizing and Removing Noise Characters from Text
T6: Custom Text Processing

Unit-3 - Sentimental Analysis **9 Hour**

Web Opinion Mining (WOM): Aspect-Based Opinion Mining- Document Level Opinion Mining. Sources for Web Opinion Mining: Blogs, News and Forums, Twitter, Other Media. Natural Language Processing: Automating Text Categorization: Using Word-Count Statistics for Categorization- Bayesian Classifier for Categorization- LSI for Categorization, Clustering Text Documents: K-means Document Clustering
T7: Supporting Indexing and Search.
T8: Using Web Scraping to Create Semantic Relations.
T9: Taking Advantage of Linked Data

Unit-4 - Web Usage Based Adaptive Systems **9 Hour**

An Overview on Web Usage Based Adaptive Systems- Web Usage Based Adaptive Systems Evolution- Process Framework: Domain Model- User Model- Adaptation Model- Approaches for Web Usage Based Systems: Content Adaptation- Navigation Adaptation- Presentation Adaptation- Privacy Issues in User Data Collection and Usage.
T10: Using RDF and RDFS Data Formats
T11: Data Quality Evaluation Tweets Performing SPARQL Queries
T12: Code Implementing SPARQL Endpoint Web Portals

Unit-5 - Recommender Systems **9 Hour**

Introduction - classification of Recommender Systems: Rule-Based Recommender Systems- Content-Based Recommender Systems- Collaborative Filtering Recommender Systems- Hybrid Recommender Systems- Sources of Knowledge for Recommender Systems: Ratings- Implicit Feedback- Social Tags- Online Social Networks – Context- Heterogeneous Sources of Knowledge.
Evaluation Metrics for Recommender Systems: Prediction-Based Metrics- Information Retrieval Related Metrics- Diversity, Novelty and Coverage-
T13: Learning user profiles
T14: Extracting Significant Information from Tweets
T15: Combining the TextResource Class with NLP

Learning Resources:

Textbooks

1. Marks Watson, “Scripting Intelligence: Web 3.0 Information Gathering and Processing”, Apress, 2009
2. Ning Zhong, Jiming Liu, Yiyu Yao, “Web Intelligence”, Springer, 2003

Reference Books

1. Juan D. Velásquez, Vasile Palade, and Lakhmi C. Jain, “Advanced Techniques in Web Intelligence-2 Web User Browsing Behaviour and Preference Analysis”
2. Shroff, Gautam, “The Intelligent Web: Search, smart algorithms, and big data”, Oxford University Press, January 2014
3. Richard Chbeir, Youakim Badr, Ajith Abraham, Aboul-Ella Hassanien, “Emergent Web Intelligence: Advanced Information Retrieval”, Springer, 2010

Wireless and Mobile Communication

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and principles of wireless communication, including the characteristics of radio frequency propagation.

2. To provide a comprehensive understanding of different mobile communication systems, from early generations (2G) to modern technologies (4G and 5G).

3. To explore the various network architectures, protocols, and standards that enable wireless and mobile communication.

4. To enable students to analyze the challenges and solutions related to mobility management, security, and quality of service in wireless networks

Course Outcomes:

CO1: To understand the basic concepts of wireless signal propagation, modulation, and multiple access techniques.
CO2: To analyze and compare different cellular network architectures and their evolution from 2G to 5G.
CO3: To comprehend the principles of mobility management and roaming in mobile networks.
CO4: To apply various wireless LAN technologies, such as Wi-Fi and Bluetooth, and their standards.
CO5: To understand the security and privacy issues in wireless and mobile communication and apply appropriate countermeasures.
CO6: To analyze and evaluate the performance of different wireless communication systems and their protocols

Course Contents:

Unit-1 - Introduction to Wireless Communication **9 Hour**

Elements of wireless communication system- Frequencies for radio communication- Signals, Noise – Types of Noise- Introduction to modulation and demodulation- Signals in the modulation- Introduction to Analog modulation schemes- Amplitude Modulation Frequency modulation- Phase Modulation Introduction to Analog modulation schemes- Amplitude Shift Keying Frequency Shift Keying Phase Shift Keying- BPSK, QPSK Multiplexing and multiple access techniques- Frequency-division multiplexing- Time-division multiplexing- Code-division multiplexing- Spread spectrum modulation- frequency hopping Spread spectrum- Direct Sequence Spread spectrum

Unit-2 - cellular Concept **9 Hour**

Cell area- Signal strength- Cell parameter- Capacity of Cell- Co channel interference- Frequency reuse- Frequency reuse- Multiple Radio access protocols Frequency division Multiple Access- Multiple Radio access protocols Frequency division Multiple Access- Multiple Access with Collision Avoidance- Space division Multiple Access Code division Multiple Access- Space division Multiple Access Code division Multiple Access- OFDM- Variants of OFDM- Comparison of Multiple Access Technique

Unit-3 – Generation of Cellular Systems **9 Hour**

Introduction to GSM- Frequency Bands and Channels- Frames in GSM- Planes and layers of GSM- Protocols- Localization and calling- Handoff – Short messaging system- GPRS EDGE- 3G CELLULAR Systems MMS- 3G CELLULAR Systems MMS- Handover- Satellite System Infrastructure- GEO, LEO, MEO- Limitations of GPS- GPS Beneficiaries of GPS- 4G Cellular systems- 4G Standards (LTE/WiMax)

Unit-4 – Communication Protocols **9 Hour**

Mobile IP- IP packet delivery- Tunneling – Reverse Tunneling- IPv6- DHCP- Tradition TCP- Congestion control- Classical TCP Snooping- Mobile TCP Fast retransmit / Fast recovery- Transaction oriented TCP TCP over 2.5/3G wireless Networks- Introduction to WAP WAP Architecture- Wireless Datagram Protocol Wireless Transaction Protocol- Wireless Session Protocol- Wireless Transport Layer Security- Wireless Markup Language- Push Architecture

Unit-5 – Protocol Architectures and Standards **9 Hour**

System Architecture- Protocol Architecture- MAC Layer and Management- 802.11a, 802.11b- HIPERLAN- Bluetooth Architecture- IEEE 802.15 IEEE 802.15.4-MANET characteristics ROUTING- AODV Routing VANET Communications in VANET- Wireless Sensor Networks- RFID Technology Two tags of RFID- Wi-Fi Standards- WiMax Standards- Fem-to-Cell Network- Push-to-talk technology for SMS

Learning Resources:

Textbooks

1. Dharma Prakash Agarwal, Qing-An Zeng, "Introduction to Wireless and Mobile Systems", CENGAGE Learning, First edition 2014
2. Jochen Schiller, "Mobile Communications", Addison Wesley, 2nd edition 2011
3. Singal TL, "Wireless Communication", Tata McGraw Hill Education Private Limited

Reference Books

1. G.I. Papadimitriou, A.S. Pomportsis, P. Nicopolitids, M.S. Obaidat, "Wireless Networks", John Wiley and Sons, 2003
2. Upena Dalal, "Wireless Communication", Oxford University Press, First edition 2009
3. Kaveh Pahlavan & Prashant Krishnamurthy, "Wireless Networks", PHI 2002
4. Martyn Mallick, "Mobile and Wireless Design Essentials", Wiley Dreamtech India Pvt. Ltd., 2014

Service Oriented Architecture

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and principles of Service-Oriented Architecture (SOA), which is a software design paradigm for building complex business applications.

2. To provide a comprehensive understanding of the key components and technologies involved in SOA, such as web services, XML, and WSDL.

3. To explore the different stages of the SOA lifecycle, including design, development, deployment, and management of services.

4. To enable students to design and implement business solutions using SOA principles and frameworks.

Course Outcomes:

CO1: To understand the basic concepts of service-oriented architecture and its role in enterprise application integration.
CO2: To comprehend the different phases of the SOA lifecycle and the best practices for service design.
CO3: To apply various web service standards and technologies like SOAP, REST, and WSDL to build services.
CO4: To analyze the security and governance issues in a service-oriented environment.
CO5: To implement and manage services using an Enterprise Service Bus (ESB).
CO6: To apply knowledge of SOA to design and develop a complete business solution

Course Contents:

Unit-1 - Introduction **9 Hour**

Introduction: SOA and MSA Basics: Service Orientation in Daily Life, Evolution of SOA and MSA. Service oriented Architecture and Microservices architecture – Drivers for SOA, Dimensions of SOA, Conceptual Model of SOA, Standards and Guidelines for SOA, Emergence of MSA Enterprise-Wide SOA: Considerations for Enterprise-wide SOA, Strawman Architecture for Enterprise-wide SOA, Enterprise SOA Reference Architecture, Object-oriented Analysis and Design (OOAD) Process, Service-oriented Analysis and Design (SOAD) Process, SOA Methodology for Enterprise

Unit-2 - Analysis and Design **9 Hour**

Service-Oriented Applications: Considerations for Service-oriented Applications, Patterns for SOA, Pattern-based Architecture for Service-oriented Applications, Composite Applications, Composite Application Programming Model. Service-Oriented Analysis and Design: Need for Models, Principles of Service Design, Nonfunctional Properties for Services, Design of Activity Services (or Business Services), Design of Data Services, Design of Client Services, Design of Business Process Services

Unit-3 - Technologies for SOA **9 Hour**

Technologies for SOA: Technologies for Service Enablement, Technologies for Service Integration, Technologies for Service Orchestration. SOA Governance and Implementation: Strategic Architecture Governance, Service Design-time Governance, Service Run-time Governance, Approach for Enterprise-wide SOA Implementation

Unit-4 - Big Data and SOA **9 Hour**

Big Data and SOA: Concepts, Big Data and its characteristics, Technologies for Big Data, Service-orientation for Big Data Solutions. Business Case for SOA: Stakeholder Objectives, Benefits of SOA, Cost Savings, Return on Investment (ROI), Build a Case for SOA

Unit-5 - SOA Best Practices **9 Hour**

SOA Best Practices: SOA Strategy – Best Practices, SOA Development – Best Practices, SOA Governance – Best Practices. EA and SOA for Business and IT Alignment: Enterprise Architecture, Need for Business and IT Alignment, EA and SOA for Business and IT Alignment

Learning Resources:

Text Books

1. Shankar Kambhampaty; Service-Oriented Architecture & Microservices Architecture: For Enterprise, Cloud, Big Data and Mobile; Wiley; 3rd Edition; 2018; ISBN: 9788126564064.
2. Thomas Erl; Service Oriented Architecture Concepts Technology & Design; Pearson Education Limited; 2015; ISBN-13: 9788131714904.

Reference Books

1. Icon Group International; The 2018-2023 World Outlook for Service-Oriented Architecture (SOA) Software and Services; ICON Group International; 1st Edition, 2017; ASIN: B06WGPN8YD.
2. Guido Schmutz, Peter Welkenbach, Daniel Liebhart; Service Oriented Architecture An Integration Blueprint; Shroff Publishers & Distributors; 2010; ISBN-13: 9789350231081.

Network Design and Management

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental principles and methodologies for designing and managing computer networks.
2. To provide a comprehensive understanding of the entire network lifecycle, from requirements analysis and design to implementation and maintenance.
3. To acquire knowledge of various network technologies and protocols and their role in a well-designed network.
4. To enable students to apply network management tools and techniques to ensure the performance, reliability, and security of a network.

Course Outcomes:

CO1: To understand the different phases of the **network design process** and the factors influencing design decisions.

CO2: To analyze and apply various **network topologies** and media types for different network environments.

CO3: To comprehend the principles of **IP addressing and subnetting** and design scalable IP addressing schemes.

CO4: To apply different **routing and switching protocols** to build a robust and efficient network infrastructure.

CO5: To understand the key principles of **network management**, including configuration, fault, and performance management.

CO6: To use network management tools to **monitor network traffic**, analyze performance, and troubleshoot common network issues

Course Contents:

Unit-1 – Introduction to Network Management **9 Hour**

Telephone Network Management, Distributed Computing Environment, TCP/IP Based Networks, Communication Protocols and Standards, Protocol Layer and Services, Challenges of IT Managers, Network Management, Network and System Management, Network Management System Platform, Current status and future of Network Management, Network Management Standards, Network Management Model - – rganizational model, Information Model, Management Information Trees, Communication Model, ASN.1, Terminology, Symbols and Conventions, Functional Model

Unit-2 – Network Protocols **9 Hour**

Introduction to SNMP, SNMP v1 model, Organization Model, System overview, SNMP v1 Information model, Structure of Management Information, Managed Objects, MIB-Object Group, System Group, Interfaces Group, Address Translation group, IP Group, ICMP Group, TCP Group, UDP Group, SNMP, MIB, User-Based Security Model, Access Control

Unit-3 – Network Monitoring Tools **9 Hour**

Remote Monitoring, RMON SMI and MIB, RMON1, RMON2, System Utilities for Management Tool, Network Statistics Measurement Systems, Traffic Load, Protocol Statistics, Data and Error Statistics, Network Management System, Components, Requirements, System Management, Network Management Applications, Configuration Management, Inventory Management, Performance Management Tools

Unit-4 – Configuration Management **9 Hour**

Network Management Applications, Fault Management - –rchitecture, Fault location, Fault isolation Algorithm, Self-healing, avoiding failures, Configuration setting, Configuration discovery and Change Control, Configuration Management Applications, Patch Management, Approaches for Performance Management, Performance Monitoring and Reporting, Performance trouble shooting, Capacity Planning, Account Management, Report Management System and User Reports, Policy Management, Service Level Management

Unit-5 – Network Design and Planning **9 Hour**

Network Design for Enterprise Network, Network Design Process, Data Collection, Data Generation, Traffic Generators, Cost Generators, Topology, Architecture, Graph, Link, Algorithms, Network Design Techniques, Performance Analysis, Queuing Essentials, Loss and Delay, Reliability, Network Cost

Learning Resources:

Textbooks

1. Greg Tomsho, Ed Tittel, David Johnson, “Guide to Network Essentials”,Eighth Edition, Cengage Learning, 2019
2. Teresa C. Piliouras, “Network Design Management and Technical Perspectives”, Second Edition, 2015
3. Mani Subramanian, “Network Management Principles and Practice”, Second Edition, Pearson Publication, 2012

Reference Books

1. Dinesh Chandra Verma, “Principles of Computer Systems and Network Management”, Springer, 2009

Advanced Data Mining

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To provide a deeper understanding of data mining techniques and algorithms beyond the introductory level.
2. To explore advanced topics such as web mining, spatial data mining, and multimedia data mining.
3. To acquire knowledge of mining complex data types and large-scale datasets, including parallel and distributed data mining.
4. To enable students to design and implement sophisticated data mining solutions for real-world applications

Course Outcomes:

CO1: To understand the principles and applications of advanced data mining models and algorithms.
CO2: To apply advanced clustering techniques to find complex patterns in high-dimensional data.
CO3: To comprehend the challenges and apply methods for mining text, web, and social network data.
CO4: To analyze and implement algorithms for mining multimedia data, including images and videos.
CO5: To understand the concepts of big data analytics and apply distributed data mining techniques using frameworks like Hadoop and Spark. CO6: To apply knowledge of data mining to solve complex problems in various domains and evaluate the performance of different models

Course Contents:

Unit-1 – Sequential Pattern Mining and Time Series Analysis **9 Hour**
Sequential Pattern Mining concepts, primitives, scalable methods; Transactional Patterns and other temporal based frequent patterns, Mining Time series Data, Periodicity Analysis for time related sequence data, Trend analysis, Similarity search in Time-series analysis.

Unit-2 – Mining Data Streams **9 Hour**
Mining Data Streams, Methodologies for stream data processing and stream data systems, Frequent pattern mining in stream data, Sequential Pattern Mining in Data Streams, Classification of dynamic data streams, Class Imbalance Problem.

Unit-3 – Graph Mining **9 Hour**
Graph Mining, Mining frequent subgraphs, finding clusters, hub and outliers in large graphs, Graph Partitioning.

Unit-4 – Web Mining **9 Hour**
Web Mining, Mining the web page layout structure, mining web link structure, mining multimedia data on the web, Automatic classification of web documents and web usage mining.

Unit-5 – Distributed Data Mining and Social Network Analysis **9 Hour**
Distributed Data Mining, Distribute data mining framework, Distributed data source, Distributed data mining techniques, Distributed classifier learning, distributed clustering, distributed association rule mining and Challenges of distributed data mining; Social Network Analysis, characteristics of social Networks.

Learning Resources:

Text Books

1. J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd Edition, Elsevier, 2011.
2. Pang-Ning Tan, M. Steinbach, and Vipin Kumar, *Introduction to Data Mining*, Addison Wesley, 2006.

Reference Books

1. G. Dong and J. Pei, *Sequence Data Mining*, Springer, 2007.

Advanced Databases

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To provide an in-depth understanding of advanced database concepts beyond the traditional relational model.

2. To explore various modern database systems, including NoSQL databases, distributed databases, and object-oriented databases.

3. To acquire knowledge of database design for performance, concurrency control, and transaction management in complex environments.

4. To enable students to analyze and select the appropriate database technology for different application requirements.

Course Outcomes:

CO1: To understand the principles and architectures of distributed database systems and their challenges.
CO2: To comprehend the concepts of transaction processing, concurrency control, and recovery in advanced database environments.
CO3: To analyze and apply the features of object-oriented and object-relational databases.
CO4: To understand and evaluate different NoSQL database models, such as key-value, document, and graph databases.
CO5: To apply data warehousing and OLAP (Online Analytical Processing) concepts for business intelligence.
CO6: To comprehend the security and privacy issues related to advanced databases and their management.

Course Contents:

Unit – I Distributed Databases
Introduction to Distributed Database Systems, Distributed Database System Architecture; Top-Down Approach, Distributed Database Design Issues, Fragmentation, Allocation, Database Integration, Bottom-up approach, Schema Matching, Schema Integration, Schema Mapping; Data and Access Control, View Management, Data Security

Unit – II Query Processing
Query processing problem, Objectives of Query processing, Complexity of Relational Algebra Operations, Characterization of Query Processors, Layers of Query Processing; Query Decomposition, Normalization, Analysis, Elimination of Redundancy and Rewriting; Localization of Distributed Data, Reduction for primary Horizontal, Vertical, derived Fragmentation; Distributed Query Execution, Query Optimization, Join Ordering, Static & Dynamic Approach, Semi-joins, Hybrid Approach

Unit – III Distributed Concurrency Control
Taxonomy of Concurrency control Mechanisms, Lock-Based Concurrency Control, Timestamp-Based Concurrency Control, Optimistic Concurrency Control, Deadlock Management; Heterogeneity issues Advanced Transaction Models, Distributed systems 2PC & 3PC protocols, Replication protocols, Replication and Failures, HotSpares

Unit – IV Parallel Databases

Introduction to Parallel Databases, Parallel Database System Architectures, Parallel Data Placement, Full Partitioning; Parallel Query Processing, Query Parallelism; Parallel Query Optimization, Search Space, Cost Model, Search Strategy; Load Balancing

Learning Resources:

Text Books

1. M. T. Özsu, Patrick Valduriez, *Principles of Distributed Database Systems*, Prentice Hall, 1999.

Reference Books

1. S. Ceri, G. Pelaggati, *Distributed Database System Principles and Systems*, McGraw-Hill, 1985.

Bio-Informatics

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and interdisciplinary nature of bioinformatics, which combines biology, computer science, and statistics.

2. To provide a comprehensive understanding of biological databases and the tools used for retrieving and analyzing biological data, such as DNA and protein sequences.

3. To explore various algorithms for sequence alignment, gene prediction, and phylogenetic analysis.

4. To equip students with the computational skills to solve biological problems, such as drug discovery and personalized medicine.

Course Outcomes:

CO1: To understand the basic concepts of molecular biology and the role of bioinformatics in modern biological research.
CO2: To apply various sequence alignment algorithms, including BLAST and FASTA, for comparing biological sequences.
CO3: To analyze and interpret the results of gene prediction and phylogenetic tree construction.
CO4: To comprehend the structure and function of biological databases, such as NCBI and PDB, and retrieve information from them.
CO5: To implement computational techniques for protein structure prediction and analysis

Course Contents:

Unit–1: Introduction to Bioinformatics
Introduction to Bioinformatics: What is a Data Base, Types of Databases, Biological Databases, Pitfalls of Biological Databases, Information Retrieval from Biological Databases

Unit–2: Pair wise Sequence Alignment
Pair wise Sequence Alignment: Evolutionary Basics, Sequence homology versus similarity, Sequence similarity versus Identity, Scoring Matrices, Statistical Significance of Sequence alignment

Unit–3: Database Similarity Searching
Database similarity searching: Unique requirement of Database searching, Heuristic Database searching, Basic alignment search tool: Comparison of FASTA and BLAST

Unit–4: Multiple Sequence Alignment and Gene Prediction
Multiple Sequence Alignment, Scoring Function, Exhaustive Algorithms, Heuristic Algorithms, Gene Prediction, Categories of gene prediction programs, Gene prediction in prokaryotes and Eukaryotes

Unit–5: Phylogenetics
Phylogenetics Basics Molecular phylogenetics and molecular basics Gene phylogeny versus species phylogeny, Forms of tree representation, Why finding a true tree is difficult, Phylogenetic tree construction methods and programs

Unit–6: Protein Structure Basics

Protein structure basics: Amino acid, peptide formation, Dihedral Angles, Hierarchy, Secondary structures, Tertiary structure, Determination of protein 3-D structure, Protein structure data base

Unit–7: Genome Mapping and Genomics

Genome mapping, assembly and comparison, Genome mapping, Genome sequencing, Genome sequence assembly, Genome Annotation, Comparative genomics, Functional Genomics, Sequence based approaches, Microarray based approaches, Comparisons of SAGE and DNA microarray

Learning Resources:

Text Books

1. Jin Xiong, *Essential Bioinformatics*, 1st Edition, Cambridge University Press, 2011.

Reference Books

1. Arthur M. Lesk, *Introduction to Bioinformatics*, 2nd Edition, Oxford University Press, 2007.

Computer Vision and Image Processing

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and techniques of image processing and computer vision.

2. To provide a comprehensive understanding of how computers can "see" and interpret digital images.

3. To explore various algorithms for tasks such as image enhancement, feature extraction, object recognition, and motion analysis.

4. To equip students with the skills to apply computer vision techniques to solve real-world problems in domains like robotics, surveillance, and medical imaging

Course Outcomes:

CO1: To understand the basic concepts of digital images, including image formation, representation, and file formats.
CO2: To apply various image enhancement techniques, such as filtering and histogram equalization, to improve image quality.
CO3: To implement different methods for image segmentation to partition an image into meaningful regions.
CO4: To extract and describe features from images using techniques like edge detection and corner detection.
CO5: To analyze and apply algorithms for object recognition and tracking in a sequence of images.
CO6: To comprehend and implement machine learning concepts for building a complete computer vision system.

Course Contents:

Unit I – Image Model and Acquisition

The image model and acquisition, image shape, sampling, intensity images, color images, range images, image capture, scanners.

Unit II – Image Operations

Statistical and spatial operations, Gray level transformations, histogram equalization, multi image operations. Spatially dependent transformations, templates and convolution, window operations, directional smoothing, other smoothing techniques.

Unit III – Segmentation and Edge Detection

Segmentation and Edge detection, region operations, Basic edge detection, second order detection, crack edge detection, edge following, gradient operators, compass & Laplace operators.

Unit IV – Morphological and Area Operations

Morphological and other area operations, basic morphological operations, opening and closing operations, area operations, morphological transformations.

Unit V – Advanced Topics

Image compression: Types and requirements, statistical compression, spatial compression, contour coding, quantizing compression. Representation and Description, Object Recognition, 3-D vision and Geometry, Digital Watermarking. Texture Analysis.

Learning Resources:

Text Books

1. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Pearson Education, 2007.
2. Milan Soanka, Vaclav Hlavac and Roger Boyle, *Digital Image Processing and Computer Vision*, Cengage Learning, 2014.

Reference Books

1. D. A. Forsyth, J. Ponce, *Computer Vision: A Modern Approach*, PHI Learning, 2009.

PE-IV (Any One FromThe Given of PE-IV)

Data Networks

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To understand the fundamental principles and architectures of data networks, which form the backbone of the internet and modern communication systems.

2. To explore the different layers of the OSI and TCP/IP models and the protocols that operate at each layer.

3. To provide a comprehensive understanding of various networking technologies, including Ethernet, Wi-Fi, and cellular networks.

4. To acquire knowledge of network performance metrics, security challenges, and management techniques

Course Outcomes:

CO1: To understand the basic concepts of data communication and the layered architecture of networks.

CO2: To analyze and implement various protocols at the data link layer, such as error control and flow control.

CO3: To comprehend the functions of the network layer, including IP addressing and routing algorithms.

CO4: To understand the principles of the transport layer, including TCP and UDP, and their applications.

CO5: To evaluate the performance of a data network using metrics like throughput and latency.

CO6: To apply knowledge of data networks to design and troubleshoot simple network topologies.

Course Contents:

Unit – 1 Physical Layer Channels
Frequency- and Time- Division Multiplexing; Error Detection; ARQ; Framing, Point-to-Point Protocols

Unit – 2 Network Layer
Error Recovery, The X.25 Network Layer Standard, The Internet Protocol

Unit – 3 Transport Layer
Transport Layer Standards, Addressing and Multiplexing TCP, Error Recovery in TCP; Flow Control in TCP/IP, Asynchronous Transfer Mode (ATM)

Unit – 4 Delay Models in Data Networks

The M/M/1 Queueing System, M/M/m, M/M/∞, M/M/m/m and other Markov Systems, Networks of Transmission Lines, Networks of Queues – Jackson's Theorem

Unit – 5 Multi-Access Communication

Packet Radio Networks, Splitting Algorithms, Carrier Sensing, Multi-access Reservations

Unit – 6 Routing in Data Networks

Wide-Area Network Routing, Interconnected Network Routing, Network Algorithms and Shortest Path Routing, Broadcasting Routing Information, Flow Models, Optimal Routing and Topological Design; Characterization of Optimal Routing

Unit – 7 Flow Control

Main Objectives of Flow Control, Window Flow Control, Rate Control Schemes, Rate Adjustment Algorithms, Classification of TCPs

Unit – 8 Software Defined Networks

Fundamental Characteristics of SDN, the OpenFlow Specification, SDN via Hypervisor-Based Overlays, SDN in the Data Center, SDN in Wide Area Networks, SDN in Mobile Networks, SDN Ecosystem and Network Virtualization

Learning Resources:

Text Books

1. Bertsekas, Dimitri, and Robert Gallager, *Data Networks* (2nd Edition). Upper Saddle River, NJ, USA, Pearson Education/Prentice Hall, 1992.
2. Walrand, Jean, and Pravin Varaiya, *High Performance Communication Networks*. Morgan Kaufmann Publishers, San Francisco, CA, 1996.

Reference Books / Online Resources

1. Stevens, W. Richard, *TCP/IP Illustrated*. Addison-Wesley Publishing Co., Reading, MA, 1994–1996.
2. Paul Goransson and Chuck Black, *Software Defined Networking: A Comprehensive Approach*. Morgan Kaufmann (Elsevier), 2014.

Design Patterns

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and principles of design patterns as a reusable solution to common problems in software design.
2. To provide a comprehensive understanding of the different categories of design patterns: creational, structural, and behavioral.
3. To explore how design patterns can improve code quality, flexibility, and maintainability.
4. To enable students to apply design patterns to solve real-world software design problems.

Course Outcomes:

CO1: To understand the role and importance of design patterns in object-oriented programming.

CO2: To apply creational design patterns to create objects in a flexible and reusable manner.

CO3: To implement structural design patterns to compose objects and classes into larger structures.

CO4: To analyze and apply behavioral design patterns to handle object interactions and responsibilities.

CO5: To compare and contrast different design patterns and select the appropriate one for a given design problem.

CO6: To apply knowledge of design patterns to design and develop robust and scalable software systems.

Course Contents:

Unit I
Introduction: What is a Design Pattern, Design Patterns in Smalltalk MVC, Describing Design Patterns, the Catalogue of Design Patterns, Organizing the catalogue, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern.

Unit II
A Case Study: Designing a Document Editor: Design Problems, Document Structure, Formatting, Embellishing the User Interface, and Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations, Spelling Checking and Hyphenation.

Unit III
Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype, Singleton.

Unit IV
Structural Pattern: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy.

Unit V

Behavioral Patterns: Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Template Method, Visitor, a Brief History, and the Pattern Community.

Learning Resources:

Text Books

1. Erich Gamma, "D“signPatterns",”Addison-Wesley, 1994.

Reference Books / Online Resources

1. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, "P“ttern-Oriented Software Architecture: A System of Pattern",”John Wiley & Sons, 1996.

Foundations of Blockchain Technology

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and principles of blockchain technology, including its underlying cryptographic and distributed systems foundations.

2. To provide a comprehensive understanding of how blockchains operate, including cryptographic hashing, digital signatures, consensus mechanisms, and smart contracts.

3. To explore the various types of blockchains, such as public, private, and consortium blockchains, and their real-world applications.

4. To enable students to analyze the security, scalability, and privacy challenges associated with blockchain technology.

Course Outcomes:

CO1: To understand the basic concepts of blockchain, including its architecture and key components.
CO2: To comprehend the role of cryptography in securing blockchain transactions and maintaining data integrity.
CO3: To analyze and compare different consensus mechanisms, such as Proof of Work and Proof of Stake.
CO4: To understand the principles of smart contracts and implement simple decentralized applications (DApps).
CO5: To explore the applications of blockchain in various industries, including finance, supply chain, and healthcare.
CO6: To analyze the scalability, security, and privacy challenges of blockchain technology.

Course Contents:

Unit-1 – Introduction to Blockchain

Introduction to Blockchain- Key Concepts of Blockchain, Features of Blockchain, Importance of Blockchain, Blockchain 1.0, 2.0, and 3.0, Issues to Centralized System, Centralized to Decentralized and Distributed System, Building Blocks of Blockchain- Distributed Ledgers & P2P Networks, Block Header, Transaction Organization.

Unit-2 – Cryptographic Primitives

Cryptographic Primitives, Basic Crypto Primitives- Hash Functions- Properties of Hash Function, Nonce, Merkle Trees, Hash Pointers, Public Key Cryptography- Public/private keys, Signature schemes, Signature correctness.

Unit-3 – Decentralization and Consensus

Decentralization- Distributed shared ledger, Distributed Consensus-Distributed Consensus Protocol. Classical theory of consensus, Byzantine Generals Problem possibility and impossibility results, Asynchronous consensus and Byzantine Fault Tolerance.

Unit-4 – Types of Blockchain (Permission-less Blockchain)

Types of Blockchain- Permission-less Blockchain- Bitcoin-Introduction to Bitcoin, Bitcoin Transaction, Bitcoin Protocol, Bitcoin Wallets, Bitcoin Block, Bitcoin Scripts, Bitcoin Network, Bitcoin Mining-Nakamoto Consensus- Proof-of-work, Mining target T, Proof-of-work equation, Mining Algorithm, Mining and reward, Block freshness, Partial and full nodes, Attacks on Bitcoin- Double-spend attacks, Selfish

mining. Ethereum Blockchain, Introduction to Ethereum, Ethereum Networks, Ethereum Wallets, Ethereum Clients, Ethereum accounts, Transactions and State, Smart contracts, Privacy-preserving smart contracts, Proof-of-stake, Variants of Ethereum blockchain.

Unit-5 – Types of Blockchain (Permissioned Blockchain)

Permissioned Blockchain- Hyperledger Fabric-, State Machine Replication, Distributed State Machines, MSP, Consensus- Raft Consensus Algorithm, Safety and liveness, Privacy based blockchain, ZCash, Zero-knowledge-proof, R3 Corda, Corda Network.

Learning Resources:

Text Books

1. A. Narayanan, J. Bonneau, E. Felten, A. Miller and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies*, Princeton University Press (PUP).
2. Imran Bashir, *Mastering Blockchain: A Deep Dive into Distributed Ledgers, Consensus Protocols, Smart Contracts, DApps, Cryptocurrencies, Ethereum, and More*, 3rd Edition, Packt Publishing, 2020, ISBN: 9781839213199.

Reference Books

1. William Magnuson, *Blockchain Democracy: Technology, Law and the Rule of the Crowd*, Cambridge University Press, 2020.
2. Pethuru Raj, Kavita Saini, Chellammal Surianarayanan, *Blockchain Technology and Applications*, CRC Press, 2021.

Information Retrieval

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To understand the fundamental concepts and techniques of Information Retrieval (IR), which is the science of searching for information from large collections of documents.

2. To explore various retrieval models, including the Boolean, vector space, and probabilistic models, and their underlying principles.

3. To provide a comprehensive understanding of the different stages of an IR system, from indexing and querying to ranking and evaluation.

4. To acquire knowledge of advanced IR topics such as relevance feedback, web search, and text classification

Course Outcomes:

CO1: To understand the basic concepts of Information Retrieval (IR) and its relationship with data mining and natural language processing.
CO2: To implement various indexing and data preprocessing techniques for building an efficient IR system.
CO3: To apply different retrieval models to rank documents based on their relevance to a user query.
CO4: To analyze and evaluate the performance of an IR system using standard metrics like precision and recall.
CO5: To comprehend the principles of relevance feedback and its role in improving search results.
CO6: To apply IR techniques to web search, text classification, and other real-world applications.

Course Contents:

Unit-1 - Introduction to Information Retrieval **9 Hour**
Information Retrieval Process, Indexing, Information retrieval model, Boolean and Vector-Space Retrieval models; Ranked retrieval; Text-similarity metrics; TF-IDF weighting; Dictionary and Postings - –okenization, stop words, Stemming, Inverted index, Skip pointers, Phrase queries.

Unit-2 - Query Evaluations and Metrics **9 Hour**
Query Expansion - relevance feedback, Rocchio algorithm, Query Expansion and its types, Query drift; XML Indexing - – vector space model for XML retrieval, Evaluation of XML retrieval, Text-centric vs. Datacentric XML retrieval; Evaluation - precision, Recall, F-measure, E-measure, Normalized recall, Evaluation problems

Unit-3 - Text Classification and Clustering **9 Hour**
Text Classification and Clustering - characterization of Text Classification – Unsupervised Algorithms: Clustering – Naïve Text Classification – Supervised Algorithms – Decision Tree – k-NN Classifier – SVM Classifier – Feature Selection or Dimensionality Reduction - –ase study using Clustering.

Unit-4 - Probabilistic and Web-Based Retrieval **9 Hour**
Probabilistic Information Retrieval - probabilistic relevance feedback, Probability ranking principle, Binary Independence Model, Bayesian network for text retrieval; Web Retrieval and Web Crawling - web search basics, crawling, indexes, Link analysis - web Characteristic, Crawling, Web As a graph, Page Rank, Hubs and Authorities - base study on Web Retrieval.

Unit-5 - Image Retrieval Methods **9 Hour**

Content Based Image Retrieval - introduction to content Based Image retrieval, Challenges in Image retrieval, Image representation, Indexing and retrieving images, Relevance feedback.

Learning Resources:

Text Books

1. C. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008.
2. Stefan Buettcher, Charles L. A. Clarke and Gordon V. Cormack, Information Retrieval: Implementing and Evaluating Search Engines, the MIT Press, 2010.
3. Tanveer Siddiqui and U. S. Tiwary, Natural Language Processing And Information Retrieval, Oxford Higher Education, 2008.
4. Ricardo Baeza-Yates, Berthier Ribeiro-Neto, "M“dern Information Retrieval: The Concepts and Technology Behind Search",”Addison Wesley, 2010.
5. ChengXiang Zhai, Sean Massung, "T“xt Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining",”ACM Books, 2016.

Reference Books

1. Soumen Chakrabarti, Mining the Web Discovering Knowledge from Hypertext Data, Morgan-Kaufmann Publishers, 2002.
2. David A. Grossman, Ophir Frieder, Information Retrieval: Algorithms and Heuristics, the Information Retrieval Series, 2nd ^{E}dition, 2004.
3. Hang Li, Learning to Rank for Information Retrieval and Natural Language Processing, Morgan & Claypool publishers, Second Edition, 2014.

Mobile Computing

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To understand the fundamental concepts and challenges of mobile computing, including the characteristics of mobile devices and networks.
2. To explore the various layers of the mobile communication protocol stack and the technologies that enable seamless connectivity.
3. To provide a comprehensive understanding of different mobile operating systems, application development frameworks, and platforms.
4. To acquire knowledge of the security, privacy, and social issues associated with mobile computing.

Course Outcomes:

CO1: To understand the basic concepts of mobile computing and its evolution.

CO2: To analyze and apply different wireless communication technologies and their standards, such as Wi-Fi, Bluetooth, and cellular networks. CO3: To comprehend the principles of mobility management and its challenges in a mobile network environment.
CO4: To understand the architecture of various mobile operating systems like Android and iOS.
CO5: To develop simple mobile applications using a relevant development framework.
CO6: To analyze the security and privacy issues in mobile computing and apply appropriate countermeasures.

Course Contents:

UNIT I
Introduction to Wireless Networks – Applications – History – Simplified Reference Model – Wireless Transmission – Frequencies – Signals – Antennas – Signal Propagation – Multiplexing – Modulation – Spread Spectrum – Cellular Systems: Frequency Management and Channel Assignment – Types of Hand-off and Their Characteristics.

UNIT II
MAC – Motivation – FDMA – TDMA – CDMA – SDMA – Telecommunication Systems – GSM: Architecture – Location Tracking and Call Setup – Mobility Management – Handover – Security – GPRS – SMS – International Roaming for GSM – Call Recording Functions – Subscriber and Service Data Management – DECT – TETRA – UMTS – IMT-2000.

UNIT III
Wireless LAN – Infrared vs. Radio Transmission – Infrastructure – Adhoc Network – IEEE 802.11 WLAN Standards – Architecture – Services – HIPERLAN – Bluetooth Architecture & Protocols.

UNIT IV
Mobile Network Layer – Mobile IP – Dynamic Host Configuration Protocol – Mobile Transport Layer – Traditional TCP – Indirect TCP – Snooping TCP – Mobile TCP – Fast Retransmit / Fast Recovery – Transmission / Time-out Freezing – Selective Retransmission – Transaction Oriented TCP.

UNIT V

WAP Model – Mobile Location Based Services – WAP Gateway – WAP Protocols – WAP User Agent Profile Caching Model – Wireless Bearers for WAP – WML – WML Scripts – WTA – iMode – SyncML.

Learning Resources:

Text Books

1. Jochen Schiller, "Mobile Communication", Second Edition, Pearson Education, 2008.
2. Theodore, S. Rappaport, "Wireless Communications, Principles, Practice", Second Edition, PHI, 2002.

Reference Books

1. William Stallings, "Wireless Communications and Networks", Second Edition, Pearson Education, 2004.
2. C. Siva Ram Murthy, B. S. Manoj, "Adhoc Wireless Networks: Architectures and Protocols", Second Edition, Pearson Education, 2008.
3. Vijay. K. Garg, "Wireless Communication and Networking", Morgan Kaufmann Publishers, 2007.

Natural Language Processing

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and techniques of Natural Language Processing (NLP), the field that enables computers to understand, interpret, and generate human language.

2. To provide a comprehensive understanding of various NLP tasks, including text preprocessing, syntactic analysis, semantic analysis, and information extraction.

3. To explore the different models and algorithms used in NLP, from traditional statistical methods to modern deep learning approaches.

4. To equip students with the skills to build NLP systems for real-world applications such as machine translation, sentiment analysis, and chatbots.

Course Outcomes:

CO1: To understand the linguistic and computational challenges of processing human language.
CO2: To apply various text preprocessing techniques, such as tokenization, stemming, and lemmatization.
CO3: To implement different models for syntactic analysis, including part-of-speech tagging and parsing.
CO4: To comprehend the principles of semantic analysis and apply techniques for word sense disambiguation and named-entity recognition.
CO5: To analyze and implement machine learning and deep learning models for NLP tasks, such as text classification and sequence-to-sequence modeling.
CO6: To apply knowledge of NLP to develop applications for tasks like sentiment analysis, text summarization, and question answering.

Course Contents:

UNIT I Lexical Analysis **9 Hour**

Lexical Analysis – Regular Expression and Automata for String Matching – Words and Word Forms – Morphology Fundamentals – Morphological Diversity of Indian Languages – Morphology Paradigms – Finite State Machine / Transducers Based Morphology – Automatic Morphology Learning – Parts of Speech – N-gram Models – Hidden Markov Models.*

UNIT II Speech Processing **9 Hour**

Biology of Speech Processing – Place and Manner of Articulation – Word Boundary Detection – Argmax based Computations – HMM and Speech Recognition – Text to Speech Synthesis – Rule based – Concatenative based Approach.*

UNIT III Parsing **9 Hour**

Theories of Parsing – Parsing Algorithms – Earley Parser – CYK Parser – Probabilistic Parsing – Resolving Attachment and Structural Ambiguity – Shallow Parsing – Dependency Parsing – Named Entity Recognition – Maximum Entropy Models – Conditional Random Fields.*

UNIT IV Lexical Knowledge Networks **9 Hour**

Meaning: Lexical Knowledge Networks – WordNet Theory – Indian Language WordNets and Multilingual Dictionaries – Semantic Roles – Word Sense Disambiguation – WSD and Multilinguality – Metaphors– Coreference and Anaphora Resolution.*

UNIT V Applications **9 Hour**

Applications: Sentiment Analysis – Text Entailment – Machine Translation – Question Answering System – Information Retrieval – Information Extraction – Cross Lingual Information Retrieval (CLIR).*

Learning Resources:

Text Books

1. Jurafsky Daniel, Martin James, "Speech and Language Processing", Second Edition, Tenth Impression, Pearson Education, 2018.
2. Christopher Manning, Schutze Heinrich, "Foundations of Statistical Natural Language Processing", MIT Press, 1999.

Reference Books

1. Allen James, "Natural Language Understanding", Second Edition, Benjamin Cumming, 1995.
2. Charniack Eugene, "Statistical Language Learning", MIT Press, 1993.

Program Analysis and Verification

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To understand the theoretical foundations and practical techniques for analyzing and verifying software programs to ensure correctness, reliability, and security.

2. To explore both static and dynamic analysis methods and their applications in identifying bugs, vulnerabilities, and performance issues.

3. To provide a comprehensive understanding of formal methods and their role in proving program properties.

4. To enable students to use state-of-the-art tools and frameworks for automated program analysis and verification.

Course Outcomes:

CO1: To comprehend the fundamental concepts of program analysis and its importance in software engineering.
CO2: To apply static analysis techniques, such as data flow analysis and control flow analysis, to understand program behavior without executing it. CO3: To implement dynamic analysis techniques to monitor program execution and identify runtime errors.
CO4: To understand the principles of model checking and apply it to verify the correctness of concurrent systems.
CO5: To analyze and apply formal methods, such as Hoare logic, for proving program properties.
CO6: To use automated tools for bug finding, security analysis, and property verification in software.

Course Contents:

Unit I
Introduction – Nature of Program Analysis, Data Flow Analysis, Equational Approach, Constraint-Based, Type and Effect Systems, Effect Systems, Algorithms.

Unit II
Data Flow Analysis – Intraprocedural Analysis, Available Expressions Analysis, Reaching Definitions Analysis, Very Busy Expressions Analysis, Live Variables Analysis, Structural Operational Semantics, Correctness of Live Variables Analysis, Monotone Frameworks, Equation Solving, Interprocedural Analysis, Shape Analysis.

Unit III
Constraint-Based Analysis – Abstract 0-CFA Analysis, Theoretical Properties, Constraint-Based 0-CFA Analysis, Adding Context Information.

Unit IV
Abstract Interpretation – Correctness, Approximation of Fixed Points, Galois Connections, Induced Operations.

Unit V
Type and Effect Systems – Control Flow Analysis, Theoretical Properties, Inference Algorithms, Effects, Behaviours. Algorithms – Worklist Algorithms, Iterating in Reverse Postorder.

Learning Resources:

Text Books

1. Flemming Nielson, Hanne R. Nielson and Chris Hankin, *Principles of Program Analysis*, Springer, 2005.
2. Edmund M. Clarke, Jr., Orna Grumberg, Daniel Kroening, Doron Peled and Helmut Veith, *Model Checking*, MIT Press, Second Edition, 2018.

Reference Books

1. Aaron R. Bradley and Zohar Manna, *The Calculus of Computation*, Springer, 2007.
2. Daniel Kroening and Ofer Strichman, *Decision Procedures: An Algorithmic Point of View*, Springer, 2008.

Software Metrics and Software Project Management

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental principles and concepts of software project management and its importance in the software development lifecycle.

2. To provide a comprehensive understanding of various software metrics for measuring and controlling the quality, cost, and schedule of a software project.

3. To explore the different stages of the project management process, including planning, risk management, and team organization.

4. To enable students to apply project management tools and techniques to effectively manage a software project from start to finish

Course Outcomes:

CO1: To understand the different phases of the software project management process and the roles of a project manager.
CO2: To apply various software sizing metrics, such as function point analysis, to estimate project size.
CO3: To analyze and apply different cost estimation models and project scheduling techniques.
CO4: To comprehend the concepts of risk management and develop strategies for identifying, analyzing, and mitigating project risks.
CO5: To understand the principles of quality management and apply metrics for measuring software quality and reliability.
CO6: To apply project management tools like Gantt charts and PERT diagrams to plan and track project progress.

Course Contents:

Unit I
Introduction to software project management: The characteristics of software projects, Reasons for IT project failure, Objectives of project management, components of Project Management, Stakeholders, Stages of Project, Activities Covered by Software Project Management, Project and Product Life Cycles, project management techniques, role of project manager.

Unit II
Software Metrics- Monitoring & measurement of SW development, cost, size and time metrics, methods and tools for metrics, issues of metrics in multiple projects.

Unit III
Project Planning: Planning process, definition, estimation, understanding organizations, stakeholder's management, project phases & project life cycles, objectives of project planning, Project schedule, Iterative steps for planning, Project Management Plan, types of risk, managing risk, risk planning and control.

Unit IV
Project Monitoring & Control: Project Control, effort data, Monitoring and Control, Quantitative techniques, Monitoring Process, Tools and techniques.

Unit V

Software Quality- Quality in SW development, quality assurance, quality standards and certifications, the process and issues in obtaining certifications, the benefits and implications for the organization and its customers, change management.
Configuration Management: Configuration management process; Software configuration items; Version control; change control; Configuration audit; Status reporting.

Unit VII
Case Studies and Practical Examples: Real-World Scenarios, Case Studies on SPM tools.

Learning Resources:

Text Books

1. Norman E. Fenton, Shari Lawrence Pfleeger, *Software Metrics - – Rigorous and Practical Approach*, 2nd Edition, PWS Pub, 1996.
2. Walker Royce, *Software Project Management*, Addison Wesley, 1998.

Reference Books

1. Bob Hughes, Mike Cotterell and Rajib Mall, *Software Project Management*, Third Edition, Tata McGraw-Hill, 2009.
2. Pankaj Jalote, *Software Project Management in Practice*, Pearson Education Inc. Delhi, 2002.

Distributed Computing

Course Code: **L:T:P: 3:0:0**

Rationale:

1. To introduce the fundamental concepts and challenges of distributed computing, where multiple computers work together to solve a problem.

2. To explore the key design issues, including inter-process communication, synchronization, and fault tolerance, in a distributed environment.

3. To provide a comprehensive understanding of different models and architectures for distributed systems, such as client-server, peer-to-peer, and grid computing.

4. To enable students to design and analyze algorithms for distributed systems and to understand their application in real-world scenarios.

Course Outcomes:

CO1: To understand the basic concepts of distributed systems, their advantages, and the challenges in their design.
CO2: To analyze and apply various inter-process communication (IPC) mechanisms for message passing in a distributed system.
CO3: To comprehend the challenges of distributed synchronization and apply solutions for mutual exclusion and deadlock handling.
CO4: To understand the principles of distributed file systems and consistency models.
CO5: To evaluate different fault tolerance and recovery techniques to build robust distributed applications.
CO6: To apply knowledge of distributed computing to design and implement simple distributed applications

Course Contents:

Unit-1 - –Introduction to Distributed Computing 9 Hour

Primitives for distributed communication, Synchronous versus asynchronous executions, Design issues and challenges, A model of distributed executions, Global state of a distributed system, Cuts of a distributed computation, A framework for a system of logical clocks, Jard–Jourdan's adaptive technique, Physical clock synchronization: NTP, Classifications and basic concepts, Complexity measures and metrics

Unit-2 - –Snapshot Recording and Graph Algorithms 9 Hour

Snapshot algorithms for FIFO channels, Variations of the Chandy–Lamport algorithm, Snapshot algorithms for non-FIFO channels Snapshots in a causal delivery system, monitoring global state, Necessary and sufficient conditions for consistent global Snapshots, Finding consistent global snapshots in a distributed computation, Elementary graph algorithms. A spanning-tree-based termination detection algorithm

Unit-3 - –Distributed Mutual Exclusion Algorithms 9 Hour

Lamport's algorithm, Ricart–Agrawala algorithm, Singhal's dynamic information-structure algorithm, Lodha and Kshemkalyani's fair mutual exclusion algorithm, Quorum-based mutual exclusion algorithms, Maekawa's algorithm, Agarwal–El Abbadi quorum-based algorithm, Token-based algorithms, Suzuki–Kasami's broadcast algorithm, Raymond's tree-based algorithm

Unit-4 - –Deadlock Detection 9 Hour

Models of deadlocks, Knapp's classification of distributed deadlock detection Algorithms, Mitchell and Merritt's algorithm for the single-resource model, Chandy–Misra–Haas algorithm for the AND model, Chandy–Misra–Haas algorithm for the OR model, Kshemkalyani–Singhal algorithm for the P-out-of-Q model

Unit-5 - Checkpointing and Rollback Recovery 9 Hour

Background and definitions, Issues in failure recovery, Checkpoint-based recovery, Log-based rollback recovery, Koo–Toueg coordinated checkpointing algorithm, Juang–Venkatesan algorithm for asynchronous checkpointing and recovery, Manivannan–Singhal quasi-synchronous checkpointing algorithm, Peterson–Kearns algorithm based on vector time, Helary–Mostefaoui–Netzer–Raynal communication-induced protocol

Learning Resources:

Textbook

1. Ajay D. Kshemkalyani, Mukesh Singhal, *Distributed Computing: Principles, Algorithms, and Systems*, Paperback, 3 March 2011

Reference Books

1. S. Tanenbaum, *Distributed Operating Systems*, Pearson Education
2. S. Tanenbaum, Maarten V. Steen, *Distributed Systems: Principles and Paradigms*, Pearson Education
3. George Coulouris, Jean Dollimore, Tim Kindberg, *Distributed Systems: Concepts and Design*

Introduction to Generative AI

Course code: **L:T:P-3:0:0**

Rationale:

This course aims to:

1. Introduce students to the fundamentals of AI, Deep Learning, and Neural Networks.
2. Explore Generative AI models such as GANs, VAEs, and Transformers.
3. Analyze the limitations of traditional RNNs and LSTMs.
4. Discuss applications, challenges, and future research directions in Generative AI.

Course Outcomes (COs)

After completing the course, students will be able to:

- **CO1:** Explain the evolution of AI and the role of deep learning in modern AI systems.
- **CO2:** Apply neural network architectures for tasks like image recognition, sequence modeling, and text generation.
- **CO3:** Perform data preprocessing and apply model training techniques for Generative AI models.
- **CO4:** Design and implement solutions using advanced neural architectures such as GANs, VAEs, and Transformers.
- **CO5:** Evaluate real-world applications of Generative AI across diverse domains like healthcare, finance, art, and natural language processing.

Content:

Module I: Foundations of AI and Neural Networks **(08 Hrs)**

History and evolution of AI and Machine Learning; the deep learning revolution; transfer learning; introduction to natural language processing; structure and anatomy of artificial neural networks; steps in training neural networks; role of parameters and hyperparameters; backpropagation algorithm and its significance.

Module II: Advanced Neural Network Architectures **(08 Hrs)**

Overview of advanced neural architectures; introduction to Generative AI models including Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Transformers; detailed study of the Attention Mechanism; Long Short-Term Memory (LSTMs) networks, their architecture, applications, and limitations.

Module III: Data Preprocessing and Model Training **(07 Hrs)**

Fundamentals of probability and statistics for AI; data preprocessing techniques such as normalization, dimensionality reduction, and feature engineering; training strategies for neural networks including batch training, stochastic gradient descent, and optimization techniques; use of regularization and dropout to avoid overfitting.

Module IV: Generative AI Applications **(07 Hrs)**

Applications of Generative AI in diverse fields including art, creativity, image and video synthesis, natural language and text generation, conversational AI, music composition, healthcare analytics, and financial modeling; case studies on real-world deployments of Generative AI models; ethical issues, risks, and challenges in implementation.

Learning Resources

Textbooks

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville – *Deep Learning*, MIT Press, 2016.
2. David Foster – *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play*, O'Reilly Media, 2nd Edition, 2022.

3. Francois Chollet – *Deep Learning with Python*, Manning Publications, 2nd Edition, 2021.

Reference Books

1. Jakub Langr and Vladimir Bok – *GANs in Action: Deep Learning with Generative Adversarial Networks*, Manning Publications, 2019.
2. Aurélien Géron – *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, O'Reilly, 3rd Edition, 2023.
3. Palash Goyal, Sumit Pandey, and Karan Jain – *Deep Learning for Natural Language Processing*, Apress, 2018.
4. Jurafsky, D., and Martin, J. H. – *Speech and Language Processing*, Pearson Education, 3rd Edition Draft, 2023.
5. Yadav, S. – *Transformers for Natural Language Processing*, Packt Publishing, 2020.

OE-I (Any one FromThe Given of OE-I)

OPEN ELECTIVE-I

IT Infrastructure Management

Course code: **L:T:P:3:0:0**

Rationale:

The course provides a comprehensive overview of the principles and practices of IT infrastructure management. It aims to equip students with the skills to plan, design, and manage the hardware, software, and network resources that form the backbone of modern organizations. The course also focuses on IT service management, security, and disaster recovery to ensure the reliability and efficiency of IT systems.

Course Outcomes:

CO1: To understand the fundamentals and components of IT infrastructure and its role in business.

CO2: To plan, design, and manage enterprise-level IT infrastructure, including data centers, networks, and servers.

CO3: To apply IT service management frameworks like ITIL to optimize service delivery.

CO4: To implement security, backup, and disaster recovery strategies for IT infrastructure.

CO5: To evaluate the performance and cost-effectiveness of IT infrastructure components.

Course Contents:

Unit – I : Configuring SSH
Hardening the SSH Server, Limiting Root Access, Configuring Alternative Ports, Modifying SELinux to Allow for Port Changes, Limiting User Access, Using Other Useful sshd Options 851, Session Options, Connection KeepAlive Options, Configuring Key-Based Authentication with Passphrases, Configuring SSH Tunnels

Unit – II : Configuring File Services
Setting Up SMB File Sharing, Installing Samba, Preparing Shared Directories on Linux, Configuring /etc/samba/smb.conf, Using Samba Users, Securing Samba, Samba-Related SELinux Parameters, Samba Firewalling, Setting Up Kerberized Samba Shares, Accessing SMB Shares, Discovering Samba Shares, Mounting Samba Shares, Performing a Multiuser Samba Mount, Mounting Samba Shares Through automount

Unit – III : Configuring NFS
Setting Up the Basic NFSv4 Server, Configuring the NFSv4 Server, Accessing NFS Shares, Testing Client Access with showmount, Making NFS Mounts Persistent, Configuring the Firewall for NFSv4, Configuring SELinux for NFSv4, Configuring NFS Kerberos Authentication, Understanding NFSv4 SELinux Transparency

Unit – IV : Configuring a Basic Web Server
Identifying the Main Configuration File, Creating Web Server Content, Understanding Apache Configuration Files, Creating Apache Virtual Hosts

Unit – V : Setting Up an SMTP Server
Understanding Email Basics, Understanding Roles in Mail Handling, The Email Transmission Process, Mail Server Solutions on RHEL, Configuring Postfix Parameters, Understanding Essential Parameters, Configuring Postfix to Relay Mail, Verifying a Working Mail Configuration

Unit – VI : Configuring DNS
Understanding DNS, The DNS Hierarchy, DNS Terminology, Understanding DNS Lookups, Understanding Resource Records, Setting Up a Cache-Only DNS Server, Understanding the Need for DNSSEC, Configuring the Unbound Caching Name Server, Configuring Trust Anchors, Troubleshooting DNS Issues, Dumping and Changing the Unbound Cache, Using dig, Analyzing Client Issues

Unit – VII : Configuring a Firewall
Understanding Linux Firewalling, Understanding Previous Solutions, Understanding Firewalld, Understanding Firewalld Zones, Understanding Firewalld Services, Working with Firewalld, Working with Firewall-cmd, Working with Firewall-config

Unit – VIII : Managing Time Synchronization
Understanding the Need for Synchronized Time, Setting Up Time Synchronization, Understanding NTP, Configuringchrony to Synchronize Time, Managing and Monitoring Time Synchronization with chronyc

Unit – IX : Configuring External Authentication and Authorization
Understanding Remote Authentication, Understanding Kerberos Basics, Understanding Kerberos Authentication, Understanding Kerberos Principals, Configuring LDAP Authentication with Kerberos Authorization, Usingnslcd or sssd as the Authentication Backend Service, Setting Up External Authentication, Using an IPA Server or Active Directory

Unit – X : Configuring Permissions
Managing File Ownership, Displaying Ownership, Changing User Ownership, Changing Group Ownership, Managing Basic Permissions, Applying Read, Write, and Execute Permissions, Managing Advanced Permissions, Managing ACLs, Understanding ACLs, Preparing Your File System for ACLs, Changing and Viewing ACL Settings with setfacl and getfacl, Working with Default ACLs, Setting Default Permissions with umask

Unit – XI : Scheduling Tasks
Configuring cron to Automate Recurring Tasks, Managing the cron Service, Understandingcron Timing, Managingcron Configuration Files, Understanding the Purpose of anacron, Managingcron Security, Configuring at to Schedule Future Tasks

Unit – XII : Disk Management
Adding Partitions, File Systems, and Persistent Mounts, Managing Swap Space

Unit – XIII : Managing Logical Volume Management (LVM) Storage
Logical Volume Management Concepts, Managing Logical Volumes, Extending Logical Volumes, Reducing Logical Volumes

Unit – XIV : Controlling and Troubleshooting the Boot Process
Selecting a Boot Target, Repairing Common Boot Issues, Resetting a Lost root Password, Repairing File System Issues at Boot, Repairing Bootloader Issues

Learning Resources:

Text Books

1. Sander van Vugt, *Red Hat RHCSA/RHCE 7 Cert Guide: Red Hat Enterprise Linux 7 (Ex200 and Ex300)*, Pearson.

Reference Books

1. Adam K. Dean, *Linux Administration Cookbook: Insightful Recipes to Work with System Administration Tasks on Linux*, Packt Publishing.

Mobile Application Development

Course code: **L:T:P:3:0:0**

Rationale:

This course is designed to introduce students to the process of developing mobile applications for various platforms. It aims to provide hands-on experience in using development tools, programming languages, and frameworks to create functional and user-friendly mobile apps. The course also covers mobile UI/UX design principles and the process of publishing an app to a store.

Course Outcomes:

CO1: To understand the mobile application development lifecycle and the characteristics of different mobile platforms.

CO2: To design and implement mobile user interfaces and user experiences that are intuitive and responsive.

CO3: To apply platform-specific programming languages and tools for developing native mobile applications.

CO4: To integrate various mobile features like sensors, location services, and databases into an app.

CO5: To test, debug, and deploy a mobile application to a public app store.

Course Contents:

Unit-I Introduction to mobile application development **9 Hour**

Introduction, Android platform: Features and architecture, versions, ART (Android Runtime), ADB (Android Debug Bridge). Development environment/IDE: Android studio and its working environment, Application anatomy: Application framework basics: resources layout, values, asset XML representation and generated R.Javafile, Android manifest file. Creating a simple application.

Unit-II GUI for android **9 Hour**

Introduction to activities life-cycle, intent filters, adding categories, linking activities, user interface design components, Views and View Groups: Basic views, picker views, adapter views, Menu, App Bar, basics of screen design; different layouts, App widgets. Lollipop Material design: new themes, new widgets, Card layouts. Recycler View, Fragments: Introduction to Fragments, life-cycle.

Unit-III Memory management **9 Hour**

Introduction to different data persistence schemes, Shared preferences, File Handling, Managing data using SQLite database. Content providers: user content provider, Android in-built content providers, Integration of social media apps.

Unit-IV Threads and android services **9 Hour**

Introduction to services – local service, remote service and binding the service, the communication between service and activity, Intent Service, Multi-Threading: Handlers, Async Task, android network programming: HTTP URL Connection, Connecting to REST-based and SOAP based Web services, Broadcast receivers: Local Broadcast Manager, Dynamic broadcast receiver, System Broadcast. Pending Intent, Notifications, Telephony Manager: Sending SMS and making calls, Interaction with server side apps.

Unit-V Building android applications **9 Hour**

Introduction to location based services, Google maps V2 services using Google API. Animations and Graphics: Property Animation, View Animations, Drawable Animations, Media and Camera API: Working with video and audio inputs, Camera API, Sensor programming: Motion sensors, Position sensors, Environmental sensors, Guidelines, policies and process of uploading apps to Google play.

Learning Resources:

Text Books

1. Dawn Griffiths, David Griffiths, *Head First: Android Development*, O'Reilly, 2015. ISBN: 9781449362188
2. Paul Deitel, Harvey Deitel, Alexander Wald, *Android 6 for Programmers: An App-Driven Approach*, Prentice Hall, 2015. ISBN: 9780134289366

Reference Books

1. Greg Milette, Adam Stroud, *Professional Android Sensor Programming*, John Wiley and Sons, Inc., 2012. ISBN: 978111265055
2. Android Developers Training. Available at: http://developer.android.com/training/index.html

System Modeling and Simulation

Course code: **L:T:P:3:0:0**

Rationale:

The course objectives are to introduce students to the principles and techniques of system modeling and simulation. It aims to provide the tools to model complex systems, simulate their behavior under different conditions, and analyze the results to make informed decisions. This course is essential for a wide range of engineering and scientific disciplines.

Course Outcomes:

CO1: To understand the concepts of system modeling and simulation and its applications.

CO2: To apply various modeling paradigms, including discrete-event, continuous, and agent-based simulation.

CO3: To use simulation software to model and analyze complex systems.

CO4: To perform statistical analysis on simulation outputs to validate models and draw conclusions.

CO5: To apply simulation techniques to solve real-world problems in domains like manufacturing, supply chain, and traffic management.

Course Contents:

Unit I

Fundamental Concepts in Mathematical Modelling: Abstraction – linearity and superposition – balance and conservation laws and the system – boundary approach.
Lumped–Element Modeling: Mechanical systems – Translational, rotational. Hydraulic systems. Thermal systems. RLC Electrical Systems.
Modeling of First–order and Second–order Systems: Governing equations for free and forced responses – transient response specifications – experimental determination – Laplace transform. Time Domain, Frequency Domain and State Space.

Unit II

Frequency response of Linear, Time invariant systems – frequency response of first–order and second–order systems – state space formulations of systems problems relating frequency response to pole location – transient response – poles and frequency response.

Unit III

Feedback systems: Systems with feedback – block diagrams – properties of feedback systems – relative stability – phase and gain margins.

Learning Resources:

Text Books

1. Cha, P. D., Rosenberg, J. J., & Dym, C. L. (2000). *Fundamentals of Modeling and Analyzing Engineering Systems*. Cambridge University.

Resources

1. Woods, R. L., & Kent, L. (1997). *Modeling and Simulation of Dynamic Systems*. Prentice Hall.
2. Mukherjee, A., & Karmakar, R. (2000). *Modeling and Simulation of Engineering Systems through Bondgraphs*. Narosa.
3. Frederick, C. (2001). *Modeling and Analysis of Dynamic Systems* (3rd ed.). Wiley.

Free and Open Source Softwares

Course code: **L:T:P:3:0:0**

Rationale:

This course is designed to provide a comprehensive understanding of Free and Open Source Software (FOSS). It aims to familiarize students with the philosophy, legal frameworks (licenses), and development models of FOSS. The course also equips students with the skills to use, contribute to, and manage projects in the open-source ecosystem.

Course Outcomes:

CO1: To understand the philosophy, history, and various licenses of Free and Open Source Software.

CO2: To use and configure a FOSS operating system and various open-source applications.

CO3: To apply version control systems like Git for collaborative development.

CO4: To contribute to an open-source project by writing code, documentation, or bug reports.

CO5: To understand the business models and community dynamics of the open-source world.

Course Contents:

Unit I : Introduction

Introduction: Open Source, Free Software, Free Software vs. Open Source software, Public Domain Software, FOSS does not mean no cost.
History: BSD, The Free Software Foundation and the GNU Project.
Methodologies: Open Source History, Initiatives, Principle and methodologies.
Philosophy: Software Freedom, Open Source Development Model.
Licenses and Patents: What Is A License, Important FOSS Licenses (Apache, BSD, GPL, LGPL), copyrights and copy lefts, Patents.
Economics of FOSS: Zero Marginal Cost, Income-generation opportunities, Problems with traditional commercial software, Internationalization.
Social Impact: Open source vs. closed source, Open source government, Open source ethics. Social and Financial impacts of open source technology, Shared software, Shared source, Open Source in Government.

Unit II : Case Studies

Example Projects: Apache web server, GNU/Linux, Android, Mozilla (Firefox), Wikipedia, Drupal, wordpress, GCC, GDB, github, Open Office.
Study: Understanding the developmental models, licensings, mode of funding, commercial/non-commercial use. Open Source Hardware, Open Source Design, Open source Teaching, Open source media, Collaboration, Community and Communication.
Contributing to Open Source Projects: Introduction to github, interacting with the community on github, Communication and etiquette, testing open source code, reporting issues, contributing code. Introduction to wi114arbonizacontributing to Wikipedia Or contributing to any prominent open source project of students choice. Starting and Maintaining own Open Source Project.

Unit III : Understanding Open Source Ecosystem

Open Source Operating Systems: GNU/Linux, Android, Free BSD, Open Solaris.
Open Source Hardware, Virtualization Technologies, Containerization Technologies: Docker, Development tools, IDEs, debuggers, Programming languages, LAMP, Open Source database technologies.

Learning Resources:

Text Books

1. Unix Concepts and Applications by Sumitabha Das, Tata McGraw Hill Education, 2006
2. The official Ubuntu Book, 8th Edition

Additional References

1. The Linux Documentation Project: http://www.tldp.org/
2. Docker Project Home: http://www.docker.com
3. Linux kernel Home:http://kernel.org
4. Open Source Initiative: https://opensource.org/
5. Linux Documentation Project:http://www.tldp.org/
6. Wikipedia: https://en.wikipedia.org/
7. https://en.wikipedia.org/wiki/Wikipedia:Contributing_to_Wikipedia
8. Github: https://help.github.com/
9. The Linux Foundation: http://www.linuxfoundation.org/

Android Development

Course code: **L:T:P:3:0:0**

Rationale:

The course provides a deep dive into the Android platform, enabling students to develop robust and feature-rich applications. It aims to cover the entire Android development lifecycle, from setting up the development environment to designing user interfaces, handling data, and deploying applications on the Google Play Store.

Course Outcomes:

CO1: To understand the Android architecture and the components of a basic Android application.

CO2: To design and implement user interfaces using various layouts and UI components.

CO3: To handle data persistence using SQLite databases, file storage, and shared preferences.

CO4: To integrate device features like sensors, location services, and camera into an app.

CO5: To test, debug, and publish an Android application on the Google Play Store.

Course Contents:

UNIT - I

Introduction to Android Operating System: Android OS design and Features – Android development framework, SDK features, Installing and running applications on Android Studio, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools
Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes
Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes

UNIT - II

Android User Interface: Measurements – Device and pixel density independent measuring
Layouts – Linear, Relative, Grid and Table Layouts
User Interface (UI) Components – Editable and non-editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers
Event Handling – Handling clicks or changes of various UI components
Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

UNIT - III

Intents and Broadcasts: Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS
Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity
Notifications – Creating and Displaying notifications, Displaying Toasts

UNIT - IV

Persistent Storage: Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory
Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference

UNIT - V

Database– Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and etindelg data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update)

Learning Resources:

Text Books

1. Professional Android 4 Application Development, Reto Meier, Wiley India (Wrox), 2012
2. Android Application Development for Java Programmers, James C. Sheusi, Cengage Learning, 2013

Reference Book

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

Data Analysis using Open-Source Tool

Course code: **L:T:P:3:0:0**

Rationale:

This course is designed to provide practical, hands-on experience in data analysis using open-source tools. It aims to equip students with the skills to load, clean, analyze, and visualize data using a tool like Python with Pandas, NumPy, and Matplotlib, or R. The course focuses on applying data analysis techniques to real-world datasets for business and scientific applications.

Course Outcomes:

CO1: To use an open-source tool like R or Python to perform data loading, cleaning, and preprocessing.

CO2: To apply descriptive statistics and data summarization techniques to understand data characteristics.

CO3: To perform exploratory data analysis (EDA) using various visualization techniques.

CO4: To implement basic machine learning models for prediction and classification.

CO5: To generate reports and dashboards to effectively communicate data analysis findings.

Course Contents:

Unit-I

Data types and analytics basics; simple linear regression; classification overview; cross-validation using validation set; introduction to decision and regression trees; R quickstart; coefficient estimation and model accuracy; k-fold cross-validation; comparison of trees and linear models.

Unit-II

R control structures; implementation of simple linear regression in R; binary and multinomial logistic regression; validation set in R; bagging and random forests; R objects and attributes; multiple linear regression and coefficients; Linear Discriminant Analysis using Bayes theorem; fitting classification trees in R.

Unit-III

Vectors and indexing; extensions of linear models; Linear Discriminant Analysis for $p > 1$; subset and stepwise selection; shrinkage methods including ridge regression; boosting; matrix operations; data visualization using matplot(); comparison of linear regression with K-Nearest Neighbors.

Unit-IV

Lasso regression and tuning; Principal Component Analysis concepts and applications; data frames; qualitative predictors; stock market example; dimension reduction and principal components regression; PCA implementation in R.

Unit-V

Lists and user-defined functions; Quadratic Discriminant Analysis in R; best-subset and stepwise selection; clustering with K-means; indexing and input/output operations in R; K-Nearest Neighbors in R; applied case study on Caravan Insurance dataset; ridge and lasso implementation in R; hierarchical clustering in R

Learning Resources:

Text Books:

1. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, *An Introduction to Statistical Learning with Applications in R*, Springer, 2013. (Covers regression, classification, cross-validation, trees, bagging, random forests, boosting, PCA, clustering, and R implementation).
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, 2nd Edition, 2009. (More advanced coverage of regression, classification, shrinkage methods, ensemble learning).
3. Norman Matloff, *The Art of R Programming*, No Starch Press, 2011. (R basics, data structures, control structures, functions, input/output).

References:

1. Jared P. Lander, *R for Everyone: Advanced Analytics and Graphics*, Addison-Wesley, 2nd Edition, 2017.
2. Peter Dalgaard, *Introductory Statistics with R*, Springer, 2nd Edition, 2008.
3. Abhijit Ghatak, *Introduction to Machine Learning with R*, Springer, 2017.
4. Brett Lantz, *Machine Learning with R*, Packt Publishing, 3rd Edition, 2019.
5. W.N. Venables and B.D. Ripley, *Modern Applied Statistics with S*, Springer, 4th Edition, 2002.
6. Randall E. Schumacker, *Learning Statistics Using R*, Sage Publications, 2014.

Concept of Industrial Electronics

Course code: **L:T:P:3:0:0**

Rationale:

The course aims to introduce the fundamental principles of industrial electronics, focusing on the design and application of electronic circuits for industrial control systems. It covers power electronics, sensors, actuators, and programmable logic controllers (PLCs), which are essential for automation and mechatronics systems.

Course Outcomes:

CO1: To understand the operation and characteristics of power electronic devices like diodes, thyristors, and MOSFETs.
CO2: To design and analyze industrial electronic circuits for power control and conversion.
CO3: To apply various sensors and actuators for industrial automation applications.
CO4: To understand the principles and programming of Programmable Logic Controllers (PLCs).
CO5: To integrate electronic components to build a basic industrial control system.

Course Contents:

UNIT – I
Review of Power Electronic Systems: Overview of Some Modern Power Semiconductor Devices.

UNIT – II
Digital Electronics: Overview, Number Systems, Integrated Circuits, Logic Families, Pin Identification.

UNIT – III
Uncontrolled rectifiers: Single phase and multiphase different circuit arrangements and their operation, analysis, performance evaluations.
Controlled rectifier: Semi Controlled and fully controlled converters, single phase and multiphase, different circuit arrangements and their operation analysis performance evaluations.

UNIT – IV
DC-DC Converters: Classification, principles of operation, step down (Buck) and step up (Boost) switched mode power supply, Buck-Boost Converter.
Inverters: Classification, theory of operation, square wave Inverter, PWM switching topology, performance evaluation, applications.

UNIT – V
Applications: DC Drives, AC Drives, Power Conditioners and Uninterruptible Power Supplies, Power Line Disturbances, Power Conditioners, UPS. Other Residential and Industrial Applications.

Learning Resources:

Textbooks:

1. B. K. Bose, *Power Electronics and AC Drives*, Prentice-Hall.
2. N. Mohan, T. M. Undeland& W. P. Robbins, *Power Electronics: Converters, Applications & Design*, John Wiley.

References:

1. L. Umanand, *Power Electronics: Essentials & Applications*, Wiley India Pvt. Ltd.

Fundamentals of Aeronautical Engineering

Course code: **L:T:P: 3:0:0**

Rationale:

This course is designed to introduce students to the foundational concepts of aeronautical engineering. It aims to provide a comprehensive overview of aerodynamics, propulsion, aircraft structures, and flight mechanics. The course serves as a gateway to understanding the principles behind aircraft design and operation.

Course Outcomes:

CO1: To understand the basic principles of flight and aerodynamic forces.
CO2: To analyze the performance characteristics of different types of aircraft.
CO3: To understand the main components and structures of an aircraft.
CO4: To comprehend the principles of jet and rocket propulsion systems.
CO5: To apply fundamental concepts of flight mechanics and stability.

Course Contents:

Unit-I History of flight **8**

Balloon flight-ornithopter-Early Airplanes by Wright Brothers, biplanes and monoplanes, Developments in aerodynamics, materials, structures and propulsion over the years.

Unit-II Aircraft configurations and its controls **10**

Different types of flight vehicles, classifications-Components of an airplane and their functions-Conventional control, powered control- Basic instruments for flying-Typical systems for control actuation.

Unit-III Basics of aerodynamics **9**

Physical Properties and structures of the Atmosphere, Temperature, pressure and altitude relationships, Newton's Law of Motions applied to Aeronautics-Evolution of lift, drag and moment. Aerofoils, Mach number, Maneuvers.

Unit -IV Basics of aircraft structures **9**

General types of construction, Monocoque, semi-monocoque and geodesic constructions, typical wing and fuselage structure. Metallic and non-metallic materials. Use of Aluminium alloy, titanium, stainless steel and composite materials. Stresses and strains-Hooke's law- stress-strain diagrams-elastic constants-Factor of Safety.

Unit-V Basics of propulsion **9**

Basic ideas about piston, turboprop and jet engines – use of propeller and jets for thrust production-Comparative merits, Principle of operation of rocket, types of rocket and typical applications, Exploration into space.

Learning Resources:

Text books

1. Anderson J.D., Introduction to Flight, McGraw-Hill, 8th edition, 2015.
2. E. Rathakrishnan, Introduction to Aerospace Engineering: Basic Principles of Flight, John Wiley, NJ, 2021.

3. Stephen A. Brandt, Introduction to Aeronautics: A Design Perspective, 2nd edition, AIAA Education Series, 2004.

References

1. Sadhu Singh, Internal Combustion Engines and Gas Turbine, SS Kataria & Sons, 2015.

2. Kermode, Flight Without Formulae, Pitman, 4th Revised edition, 1989.

Remote Sensing Concepts

Course code: **L:T:P: 3:0:0**

Rationale:

The course objectives are to introduce students to the fundamental principles of remote sensing. It aims to provide knowledge of the electromagnetic spectrum, satellite systems, and image processing techniques for environmental, urban, and agricultural applications.

Course Outcomes:

CO1: To understand the basic principles of remote sensing and the electromagnetic spectrum.

CO2: To identify and differentiate various types of remote sensing platforms and sensors.

CO3: To apply image processing techniques for the analysis of remote sensing data.

CO4: To interpret remote sensing imagery for applications in land use, agriculture, and water resource management.

CO5: To understand the integration of remote sensing data with GIS for spatial analysis.

Course Contents:

Unit-I Remote sensing and electromagnetic radiation **9**

Definition– components of RS – History of Remote Sensing – Merits and demerits of data collation between conventional and remote sensing methods - Electromagnetic Spectrum – Radiation principles - Wave theory, Planck‘s law, Wien’s Displacement Law, Stefan’s Boltzmann law, Kirchoff‘s law – Radiation sources: active & passive - Radiation Quantities

Unit-II EMR interaction with atmosphere and earth material **9**

Standard atmospheric profile – main atmospheric regions and its characteristics – interaction of radiation with atmosphere – Scattering, absorption and refraction – Atmospheric windows - Energy balance equation – Specular and diffuse reflectors – Spectral reflectance & emittance – Spectroradiometer– Spectral Signature concepts – Typical spectral reflectance curves for vegetation, soil and water – solid surface scattering in microwave region.

Unit-III Orbits and platforms **9**

Motions of planets and satellites – Newton‘s law of gravitation - Gravitational field and potential - Escape velocity - Kepler‘s law of planetary motion - Orbit elements and types – Orbital perturbations and maneuvers – Types of remote sensing platforms - Around based, Airborne platforms and Space borne platforms – Classification of satellites – Sun synchronous and Geosynchronous satellites – Lagrange Orbit.

Unit-IV Sensing techniques **9**

Classification of remote sensors – Resolution concept : spatial, spectral, radiometric and temporal resolutions - Scanners - Along and across track scanners – Optical-infrared sensors – Thermal sensors – microwave sensors – Calibration of sensors - High Resolution Sensors - –IDAR , UAV – Orbital and sensor characteristics of live Indian earth observation satellites.

Unit-V Data products and interpretation **9**

Photographic and digital products – Types, levels and open source satellite data products – selection and procurement of data – Visual interpretation: basic elements and interpretation keys - Digital interpretation – Concepts of Image rectification, Image enhancement and Image classification.

Learning Resources:

Textbooks

1. Thomas M. Lillesand, Ralph W. Kiefer and Jonathan W. Chipman, Remote Sensing and Image Interpretation, John Wiley and Sons, Inc, New York, 2015.

2. George Joseph and C. Jeganathan, Fundamentals of Remote Sensing, Third Edition, Universities Press (India) Private Limited, Hyderabad, 2018.

References

1. Janza F.Z., Blue H.M. and Johnson J.E., Manual of Remote Sensing, Vol.1, American Society of Photogrammetry, Virginia, USA, 2002.

2. Verbyla David, Satellite Remote Sensing of Natural Resources, CRC Press, 1995.

3. Paul Curran P.J., Principles of Remote Sensing, Longman, RLBS, 1988.

4. Charles Elachi and Jacob Van Zyl, Introduction to Physics and Techniques of Remote Sensing, 2nd ^{E}dition, Wiley Publication, 2006.

5. Basudeb Bhatta, Remote Sensing and GIS, Oxford University Press, 2011.

Urban Agriculture

Course code: **L:T:P: 3:0:0**

Rationale:

This course is designed to introduce students to the principles and practices of urban agriculture. It aims to provide knowledge of sustainable food production in urban environments, covering topics like vertical farming, hydroponics, and community gardens. The course also explores the economic and social benefits of urban food systems.

Course Outcomes:

CO1: To understand the concepts and importance of urban agriculture for sustainable food systems.

CO2: To design and implement various urban farming systems, such as container gardening and rooftop farms.

CO3: To apply advanced techniques like hydroponics and aquaponics for food production.

CO4: To analyze the economic viability and social impact of urban agriculture projects.

CO5: To understand the role of technology and innovation in urban food production.

Course Contents:

Unit-I Introduction **9**

Benefits of urban agriculture - Economic benefits, environmental benefits, social and cultural benefits, educational, skill-building and job training benefits, health, nutrition and food accessibility benefits.

Unit-II Vertical farming **9**

Vertical farming - Types, green faIliving/green wall - Modular green wall, vegetated mat wall - Structures and components for green wall system: plant selection, growing media, irrigation and plant nutrition: design, light, benefits of vertical gardening. Roof garden and its types. Kitchen garden, hanging baskets: the house plants/indoor plants.

Unit-III Soil less cultivation **9**

Hydroponics, aeroponics, aquaponics: merits and limitations, costs and challenges, backyard gardens, tactical gardens, street landscaping, forest gardening, greenhouses, urban beekeeping.

Unit-IV Modern concepts **9**

Growth of plants in vertical pipes in terraces and inside buildings, micro irrigation concepts suitable for roof top gardening, rain hose system, greenhouse, polyhouse and shade net system of crop production on roof tops.

Unit-V Waste management **9**

Concept, scope and maintenance of waste management - Recycle of organic waste, garden wastes, solid waste management, scope, microbiology of waste, other ingredients like insecticide, pesticides and fungicides residues, waste utilization.

Learning Resources:

Textbooks

1. Martellozzo F. and J. S. Landry, Urban Agriculture, Scitus Academics LLC, 2020.

2. Rob Roggema, Sustainable Urban Agriculture and Food Planning, Routledge Taylor and Francis Group, 2016.

3. Akrong M. O., Urban Agriculture, LAP Lambert Academic Publishing, 2012.

References

1. Agha Rokh A., Evaluation of ornamental flowers and fishes breeding in Bushehr urban wastewater using a pilot-scale aquaponic system, Water and Wastewater, 19 (65): 47–53, 2008.

2. Agrawal M., Singh B., Rajput M., Marshall F. and Bell J. N. B., Effect of air pollution on peri-urban agriculture: A case study, Environmental Pollution, 126 (3): 323–329, 2003.Jac Smit and Joe Nasr, Urban agriculture for sustainable cities: using wastes and idle land and water bodies as resources, Environment and Urbanization, 1992.

Drinking Water Supply and Treatment

Course code: **L:T:P:3:0:0**

Rationale:

The course aims to provide a comprehensive understanding of the principles and technologies for drinking water supply and treatment. It covers the sources of water, water quality parameters, and various treatment processes to ensure the safety and potability of water. The course also addresses the challenges of water scarcity and sustainable management.

Course Outcomes:

CO1: To understand the sources of drinking water and the importance of water quality.
CO2: To analyze and interpret water quality parameters based on national and international standards.
CO3: To design and evaluate various water treatment processes, including coagulation, filtration, and disinfection.
CO4: To understand the principles of water distribution systems and network design.
CO5: To manage and solve problems related to drinking water supply and treatment in a sustainable manner.

Course Contents:

UNIT I SOURCES OF WATER (9)

Public water supply system – Planning, Objectives, Design period, Population forecasting; Water demand – Sources of water and their characteristics, Surface and Groundwater – Impounding Reservoir – Development and selection of source – Source Water quality – Characterization – Significance – Drinking Water quality standards.

UNIT II CONVEYANCE FROM THE SOURCE (9)

Water supply – intake structures – Functions; Pipes and conduits for water – Pipe materials – Hydraulics of flow in pipes – Transmission main design – Laying, jointing and testing of pipes – appurtenances– Types and capacity of pumps – Selection of pumps and pipe materials.

UNIT III WATER TREATMENT (9)

Objectives – Unit operations and processes – Principles, functions, and design of water treatment plant units, aerators of flash mixers, Coagulation and flocculation – sand filters – Disinfection – Construction, Operation and Maintenance aspects.

UNIT IV ADVANCED WATER TREATMENT (9)

Water softening – Desalination– R.O. Plant – demineralization – Adsorption – Ion exchange – Membrane Systems – Iron and Manganese removal – Defluoridation – Construction and Operation and Maintenance aspects.

UNIT V WATER DISTRIBUTION AND SUPPLY (9)

Requirements of water distribution – Components – Selection of pipe material – Service reservoirs– Functions – Network design – Economics – Computer applications – Appurtenances– Leak detection – Principles of design of water supply in buildings – House service connection – Fixtures and fittings, systems of plumbing and types of plumbing.

Learning Resources:

Text Books:

1. Garg. S.K., *Water Supply Engineering*, Khanna Publishers, Delhi, September 2008.
2. Punmia B.C., Arun K. Jain, Ashok K. Jain, *Water Supply Engineering*, Lakshmi Publication Private Limited, New Delhi, 2016.
3. Rangwala, *Water Supply and Sanitary Engineering*, February 2022.
4. Birdie. G.S., *Water Supply and Sanitary Engineering*, Dhanpat Rai and Sons, 2018.

References:

1. Fair. G.M., Geyer. J.C., *Water Supply and Wastewater Disposal*, John Wiley and Sons, 1954.
2. Babbit. H.E., Donald. J.J., *Water Supply Engineering*, McGraw Hill Book Co, 1984.
3. Steel. E.W. et al., *Water Supply Engineering*, McGraw Hill International Book Co, 1984.
4. Duggal. K.N., *Elements of Public Health Engineering*, S. Chand and Company Ltd, New Delhi, 1998.

Electric Vehicle technology

Course code: **L:T:P:3:0:0**

Rationale:

This course is designed to introduce students to the core technologies of electric vehicles (EVs). It aims to provide a detailed understanding of electric vehicle components, battery management systems, and charging infrastructure. The course also covers the performance characteristics and control strategies of EVs.

Course Outcomes:

CO1: To understand the fundamental components and architecture of electric vehicles.

CO2: To analyze the performance and characteristics of different types of batteries for EVs.

CO3: To apply power electronic converters for motor control and battery charging.

CO4: To understand the principles of battery management systems (BMS) and charging standards.

CO5: To evaluate the environmental and economic impact of electric vehicles.

Course Contents:

Unit-I Rotating power converters **9**

Magnetic circuits – DC machine and AC machine – Working principle of generator and motor – DC and AC – Voltage and torque equations – Characteristics and applications. Working principle of special machines like brushless DC motor, switched reluctance motor and PMSM.

Unit-II Static power converters **9**

Working and characteristics of power diodes, MOSFET and IGBT. Working of uncontrolled rectifiers, controlled rectifiers (single phase and three phase), DC choppers, single and three phase inverters, multilevel inverters and matrix converters.

Unit-III Control of DC and AC motor drives **9**

Speed control for constant torque, constant HP operation of all electric motors – DC/DC chopper based four quadrant operation of DC motor drives, inverter based V/f operation (motoring and braking) of induction motor drives, transformation theory, vector control operation of induction motor and PMSM, brushless DC motor drives, switched reluctance motor drives.

Unit-IV Hybrid electric vehicle architecture and power train components **9**

History of evolution of electric vehicles – Comparison of electric vehicles with internal combustion engines – Architecture of electric vehicles (EV) and hybrid electric vehicles (HEV) – Plug-in hybrid electric vehicles (PHEV) – Power train components and sizing, gears, clutches, transmission and brakes.

Unit-V Mechanics of hybrid electric vehicles and control of vehicles **9**

Fundamentals of vehicle mechanics – Tractive force, power and energy requirements for standard drive cycles of HEVs – Motor torque and power rating and battery capacity. HEV supervisory control – Selection of modes – Power split mode – Parallel mode – Engine brake mode – Regeneration mode – Series parallel mode.

Learning Resources:

Text Books

1. Stephen D. Umans, *Fitzgerald & Kingsley's Electric Machinery*, 7th Edition, Tata McGraw Hill, 2020.

2. Rashid M. H., *Power Electronics: Circuits, Devices and Applications*, 4th Edition, 10th Impression, Pearson, 2021.

3. Paul C. Krause, Oleg Wasynczuk, Scott D. Sudhoff, Steven D. Pekarek, *Analysis of Electric Machinery and Drive Systems*, 3rd Edition, Wiley-IEEE Press, 2013.

Reference Books

1. Bogdan M. Wilamowski, J. David Irwin, *The Industrial Electronics Handbook, Second Edition: Power Electronics and Motor Drives*, CRC Press, 2011.

2. Iqbal Husain, *Electric and Hybrid Electric Vehicles*, CRC Press, 2021.

3. Wei Liu, *Hybrid Electric Vehicle System Modeling and Control*, 2nd Edition, Wiley, 2017.

4. James Larminie, John Lowry, *Electric Vehicle Technology Explained*, 2nd Edition, Wiley, 2012.

Introduction to PLC Programming

Course code: **L:T:P:3:0:0**

Rationale:

The course objectives are to introduce students to the principles and programming of Programmable Logic Controllers (PLCs). It aims to provide hands-on experience in using PLC software to develop logic for industrial automation and control systems.

Course Outcomes:

CO1: To understand the architecture and components of a Programmable Logic Controller (PLC).
CO2: To apply ladder logic and other programming languages for PLC-based control.
CO3: To design and implement control systems for industrial automation using a PLC.
CO4: To test, debug, and troubleshoot PLC programs.
CO5: To understand the applications of PLCs in manufacturing, process control, and robotics.

Course Contents:

Unit-I Introduction to plc **9**

introduction to plc: microprocessor, i/o ports, isolation, filters, drivers, microcontrollers/dsp, plc/ddc- plc construction: what is a plc, plc memories, plc i/o, , plc special i/o, plc types.

Unit-II PLC instructions **9**

plc basic instructions: plc ladder language- function block programming- ladder/function block functions- plc basic instructions, basic examples (start stop rung, entry/reset rung)- configuration of sensors, switches, solid state relays-interlock examples- timers, counters, examples.

Unit-III PLC programming **9**

different types of plc program, basic ladder logic, logic functions, plc module addressing, registers basics, basic relay instructions, latching relays, arithmetic functions, comparison functions, data handling, data move functions, timer-counter instructions, input-output instructions, sequencer instructions

Unit-IV Communication of PLC and SCADA **9**

Communication protocol– modbus, hart, pr131arbonizcommunication facilities scada: - hardware and software, remote terminal units, master station and communication architectures

Unit-V Case studies **9**

Stepper motor control- elevator control-CNC machine control- conveyor control-interlocking problems

Learning Resources:

Text books

1. Frank petruzzula, programmable logic controllers, tata mc-graw hill edition
2. John w. webb, ro131arbon. reis, programmable logic controllers principles and applications, phi publication

References

1. Madhuchanndmitra and samerjit se131arbonizprogrammable logic controllers industrial automation an introduction, penram international publishing pvt. ltd.

2. J. R. Hackworth and F. D. Hackworth, programmable logic controllers principles and applications, pearson publication

Nano Technology

Course code: **L:T:P: 3:0:0**

Rationale:

The course aims to introduce the fundamental concepts of nanotechnology, its principles, and applications. It provides a comprehensive overview of nanomaterials, their synthesis, characterization, and their unique properties. The course also explores the potential of nanotechnology in various fields like medicine, electronics, and energy.

Course Outcomes:

CO1: To understand the fundamental concepts and principles of nanotechnology.
CO2: To describe the synthesis methods and characterization techniques for nanomaterials.
CO3: To analyze the unique properties of nanomaterials at the nanoscale.
CO4: To apply nanotechnology principles in the design of functional materials and devices.
CO5: To understand the ethical, social, and environmental implications of nanotechnology.

Course Contents:

Unit-I Introduction **8**

General definition and size effects–important nano structured materials and nano particles-importance of nano materials- Size effect on thermal, electrical, electronic, mechanical, optical and magnetic properties of nanomaterials- surface area - –and gap energy and applications. Photochemistry and electrochemistry of nanomaterials – ionic properties of nanomaterials- nano catalysis.

Unit-II Synthesis of nanomaterials **8**

Bottom up and top-down approach for obtaining nano materials - –recipitation methods – sol gel technique – high energy ball milling, CVD and PVD methods, gas phase condensation, magnetron sputtering and laser deposition methods – laser ablation, sputtering.

Unit-III Nano composites **10**

Definition- importance of nanocomposites- nano composite materials-classification of composites- metal/metal oxides, metal-polymer- thermoplastic based, thermoset based and elastomer based-influence of size, shape and role of interface in composites applications.

Unit-IV Nano structures and characterization techniques **10**

Classifications of nanomaterials - –ero dimensional, one-dimensional and two-dimensional nanostructures-kinetics in nanostructured materials- multilayer thin films and superlattice-clusters of metals, semiconductors and nanocomposites. Spectroscopic techniques, diffraction methods, thermal analysis method, BET analysis method.

Unit-V Applications of nano materials **9**

Overview of nanomaterials properties and their applications, nano painting, nano coating, nanomaterials for renewable energy, molecular electronics and nanoelectronics – nanobots-biological applications. Emerging technologies for environmental applications- practice of nanoparticles for environmental remediation and water treatment.

Learning Resources:

Text books

1. Mick Wilson, Kamali Kannangara, Geoff Smith, Michelle Simmom, Burkhard Raguse, “Nano Technology: Basic Science & Engineering Technology”, 2005, Overseas Press
2. G. Cao, “Nanostructures & Nanomaterials: Synthesis, Properties & Applications” Imperial College Press, 2004
3. William A Goddard “Handbook of Nanoscience, Engineering and Technology”, 3rd Edition, CRC Taylor and Francis group 2012

References

1. R.H.J.Hannink&A.J.Hill, Nanostructure Control, Wood Head Publishing Ltd., Cambridge, 2006
2. C.N.R.Rao, A.Muller, A.K.Cheetham, The Chemistry of Nanomaterials: Synthesis, Properties and Applications Vol. I & II, 2nd edition, 2005, Wiley VCH Verlag Gibtl& Co
3. Ivor Brodie and Julius J. Muray, “The Physics of Micro/Nano – Fabrication”, Springer International Edition, 2010

Functional Materials

Course code: **L:T:P:3:0:0**

Rationale:

The course is designed to introduce students to the principles and applications of functional materials. It aims to provide knowledge of materials with unique electronic, magnetic, optical, and thermal properties that are essential for modern technologies and engineering applications.

Course Outcomes:

CO1: To understand the fundamental properties and classifications of functional materials.
CO2: To describe the structure-property relationships of different functional materials.
CO3: To apply functional materials in the design of electronic, optical, and energy-related devices.
CO4: To understand the synthesis and characterization techniques for functional materials.
CO5: To analyze the performance of functional materials for specific engineering applications.

Course Contents:

Unit I

The Origin of Functional Materials (FMs), Potential Applications of FMs, Classification of FMs, Processing Techniques: Powder Metallurgy Route, Melt-processing Route, Vacuum arc melting, Vacuum induction melting, Vapor deposition and types

Unit II

Specific properties of functional materials: Magnetic materials, Electronic Materials and Sensors, Electric Contact Materials, Conducting Thermoplastics and polymer composites, Surface coatings for functional applications, Biomaterials and Shape memory metals, Invar alloys

Unit III

Batteries and fuel cell: Solar energy harvesting, Reflective and anti-reflective layers, Waste heat recovery materials

Unit IV

Microstructure-property correlations: characteristic dimensions and spatial variations, volume fraction, rule of mixtures and effective field parameters; characterization of FMs, macrostructural thermomechanical properties, material properties of ceramic-metal FMs, basic mathematical modelling

Learning Resources:

Text Books

1. *Engineering Materials for Technological Needs, Vol. 2- Functional Materials: Electrical, Dielectric, Electromagnetic, Optical and Magnetic Applications* – D.D.L. Chung, World Scientific Publishing, 2010.
2. *Composite Materials- Functional Materials for Modern Technologies* – D.D.L. Chung, Springer, 2002.
3. *Functionally Graded Materials- Design, Processing and Applications* – Y. Miyamoto, W.A. Kaysser, B.H. Rabin, A. Kawasaki, R.G. Ford (Eds.), Springer, 1999.

Reference Books

1. *Functionally Graded Materials - –onlinear Analysis of Plates and Shells* – Hui-Shen Shen, CRC Press, 2009.

Other Suggested Readings

1. https://nptel.ac.in/courses/113/105/113105081/
2. https://www.researchgate.net/publication/227194436_Functional_Materials_Properties_Processing_and_Applications
3. https://www.intechopen.com/books/7554

Traditional Indian Foods

Course code: **L:T:P:3:0:0**

Rationale:

The course aims to introduce students to the science and technology behind traditional Indian foods. It covers the preparation, processing, and preservation techniques of various regional foods. The course also explores the nutritional value and cultural significance of traditional Indian cuisine.

Course Outcomes:

CO1: To understand the principles and methods of preparing traditional Indian foods.
CO2: To analyze the nutritional composition and health benefits of various traditional foods.
CO3: To apply food processing and preservation techniques to traditional Indian foods.
CO4: To understand the microbial and enzymatic changes during food fermentation.
CO5: To appreciate the cultural significance and regional diversity of traditional Indian cuisine.

Course Contents:

Unit I

Indian traditional food: Introductory remark, history of Indian food culture and traditional foods, Indian regional food products, role of traditional food processing technologies in India (3 Lectures)

Unit II

Traditional food product technology: Significance & characteristics of traditional food products processing & preservation techniques, soaking, germination, dehulling, sun drying, osmotic drying, brining, pickling & smoking (4 Lectures)
Health aspects of traditional foods: Traditional Vs fast foods- Nutritional importance, food safety, bioactive components & cost (2 Lectures)

Unit III

Fermentation technology: Fermentation process, types, advantages, fermentation media, inoculum preparation, upstream and downstream fermentation process, types of fermenters (3 Lectures)
Traditional and oriental fermented food products: Idli, Dhokla, Tempeh, Miso, Natto, AngKak, Saurkraut, Fermented dairy products (2 Lectures)

Unit IV

Indian traditional foods: Traditional foods based on cereals, legumes, oilseed, fruits, vegetables, dairy and spices (4 Lectures)
Process Technology for convenience traditional food products: Ready to cook (Instant mixes), Ready to eat, Traditional beverages. (3 Lectures)

Unit V

Traditional food as functional food: Ayurvedic concept in food & nutrition, Need of Ayurvedic foods, Ayurvedic nutritional supplements & food products, Role of traditional and ayurvedic foods in health promotion and disease prevention, Application of probiotics & prebiotics in traditional food, Traditional Indian foods as providers of polyphenols & dietary fibers (4 Lectures)
Consumer preferences and innovation in traditional foods (2 Lectures)

Unit VI

Packaging & marketing of traditional food product: Packaging aspects of traditional foods, Opportunities and challenges for commercialization of traditional foods at national & international market. (3 Lectures) Food safety & regulatory aspect of traditional food (2 Lectures)

Learning Resources:

Text Books

1. Indian Food Science: A Health and Nutrition Guide to Traditional Recipes – Ruth N. Davidar
2. Traditional Indian Functional Foods – Krishnapura Srinivasan
3. Traditional Cuisines of India – Indian Trust for Rural Heritage and Development Publication

Reference Books

1. Industrial Microbiology – A.H. Patel
2. Food Microbiology – W.C. Frazier

Introduction to food processing

Course code: **L:T:P: 3:0:0**

Rationale:

This course is designed to introduce students to the fundamental principles and technologies of food processing. It aims to provide an understanding of the various methods used to preserve, package, and enhance the safety and quality of food products.

Course Outcomes:

CO1: To understand the basic principles and objectives of food processing.
CO2: To describe the different methods of food preservation, including thermal processing, refrigeration, and drying.
CO3: To apply food packaging technologies to maintain product quality and shelf life.
CO4: To analyze food safety and quality control measures in the food processing industry.
CO5: To understand the role of food additives and their impact on food quality.

Course Contents:

UNIT – I

Introduction: Defining food; Classification of food; Constituents of foods; Food processing; Food preservation; Food Spoilage – Introduction, Causes of food spoilage, Food-borne intoxication, Food-borne infection

UNIT – II

Food poisoning, Food Preservation and Processing: Introduction; necessary; Methodology; Principles and Methods of food preservation, High Temperature Preservation: Introduction; Blanching; Pasteurization; Sterilization; Canning, Drying, Dehydration and Concentration: Introduction; Purpose; Water activity and relative humidity; Factors affecting rate of drying and dehydration; Drying methods; Changes during drying and dehydration; different Driers; Concentration- Methods of concentration, Changes; Effect of drying, dehydration and concentration on quality of foods, Food Irradiation: Introduction; Radiation sources; Measurement of radiation dose; Mechanism of Action; Type of irradiation; Factors affecting food irradiation; Effect of irradiation

UNIT – III

Preservation using Sugar, Salt and Acids: Sugar – Introduction, Factors affecting osmotic pressure of sugar solution, Foods preserved using sugar; Salt: Introduction, Antimicrobial activity of salt, Estimation of salt, Food products preserved using salt; Acid – Introduction, Mechanism, Common foods preserved using acids, Preservation by Use of Chemical preservatives: Introduction; Objectives; Factors affecting antimicrobial activity of preservatives; Type of chemical preservatives; Sulphur dioxide, Benzoic acid, etc.; Use of other chemicals like acidulates, antioxidants, mold inhibitors, antibodies, etc. Food Fermentation: Introduction, methods, common fermented foods

UNIT – IV

Recent methods in Processing: Introduction; PEF, HPP, Ultrasound, Dielectric heating; Microwave heating, Osmic heating; Infrared heating; UV light, X-rays, Membrane processing, Ionization; High intensity electric field in pulses; New hybrid drying technologies; Monitoring by NMR and MRI Technology, etc.

UNIT – V

Effect of processing on nutritional value of food: Introduction; Consuming raw foods; Changes during meat grilling; Effect of processing on vitamins; Effect of processing on minerals; Effect of processing on carbohydrates; Effect of processing on lipids

Learning Resources:

Text Books

1. Preservation of Fruits & Vegetables – Girdhari Lal, G. S. Siddappa, G. L. Tandon, Indian Council of Agricultural Research Publications, 1986
2. Food Processing Technology: Principles and Practice – P. Fellows, CRC Press, 2000, ISBN: 9780849308871
3. Introduction to Food Processing – Jelen P., Prentice Hall, 1985

Reference Books

1. Handbook of Food Preservation – Shafiur Rahman M., CRC Press, 2007, ISBN: 9781420017373
2. Emerging Technologies for Food Processing – Da-Wen Sun, Academic Press, 2005, ISBN: 9780080455648
3. Handbook of Analysis and Quality Control for Fruit and Vegetable Products – Ranganna S., 2nd Edition, Tata-McGraw-Hill, 2001

IPR for Pharma Industry

Course code: **L:T:P:3:0:0**

Rationale:

The course aims to provide a comprehensive understanding of Intellectual Property Rights (IPR) as they apply to the pharmaceutical industry. It covers the principles of patents, trademarks, and copyrights in the context of drug discovery, development, and commercialization. The course also addresses the legal and ethical issues related to IPR in the pharma sector.

Course Outcomes:

CO1: To understand the basic concepts of Intellectual Property Rights (IPR) and their relevance in the pharmaceutical industry.
CO2: To apply the principles of patent law to protect new drug discoveries and formulations.
CO3: To analyze the legal and regulatory frameworks for trademarks and copyrights in the pharma sector.
CO4: To understand the process of patent filing, searching, and infringement analysis.
CO5: To evaluate the strategic importance of IPR in the business and competitive landscape of the pharmaceutical industry.

Course Contents:

Unit I

History of Pharmaceutical Industry, Law and Pharmaceutical Industry, Protection of medical Inventions, Protection of Consumers and Families, IPRs involved in Pharma Industry: Patents, Industrial designs, Trademarks, Copyright, Trade Secrets

Unit II

Patent and Pharma Industry, Patents relating to pharmaceutical inventions, Patents relating to formulations and compositions, Criteria of Patentability, Impact of WTO on Pharmaceutical Patents

Unit III

Trademark and Pharma Industry, Pharma Trademark infringement and passing off, The concept of colour as a Mark in Pharma Industry, Phonetic Similarity and Pharma Industry, Challenges of Trademark Management in Pharma Industry

Unit IV

Copyright and Pharma Industry, Copyright License for Pharmaceutical Industry, Industrial Design and Pharma Industry, Digital design for pharmaceutical product and process development, Trade secrets in Pharma Industry

Learning Resources:

Textbooks

1. Feroz Ali Khader (2011). *Law of Patents WithA Special Focus On Pharmaceuticals In India*. Lexis Nexis.

2. Bayya Subba Rao and P. V. Appaji (2019). *Intellectual Property Rights in Pharmaceutical Industry: Theory and Practice*. BSP Books.

References

1. G.G. Nair (2008). Impact of TRIPS on Indian Pharmaceutical Industry. *Journal of Intellectual Property Rights*, Vol. 13, Sept., pp. 432–441.

2. Damodaran, A.D. (2008). Indian Patent Law InThe Post TRIPS Decade: S&T Policy Appraisal. *NISCAIR*, Sept.

3. Chaudhuri, Sudip (2005). *The WTO and India's Pharmaceuticals Industry: Patent Protection TRIPS and Developing Countries*. Oxford University Press, New Delhi.

4. Griliches, Z. (1990). Patent statistics as economic indicators: a survey. *Journal of Economic Literature*, Vol. 28(4), 1661–1707.

5. Dey, S. (2016). The new national IPR Policy 2016: implications for the pharmaceutical industry. *Gggi Management Review.*

6. Mitsumori, Yaeko (2018). *The Indian Pharmaceutical Industry: Impact of Changes in the IPR Regime*. Springer.

Basics of Textile Finishing

Course code: **L:T:P:3:0:0**

Rationale:

This course is designed to introduce students to the fundamental principles and processes of textile finishing. It aims to provide knowledge of the various finishing techniques used to enhance the appearance, performance, and functionality of textile materials.

Course Outcomes:

CO1: To understand the purpose and importance of textile finishing processes.
CO2: To describe the different types of chemical and mechanical finishing methods.
CO3: To apply finishing techniques to improve properties like durability, softness, and water repellency.
CO4: To analyze the effects of finishing processes on the physical and chemical properties of textiles.
CO5: To evaluate the quality of finished textile products using standard testing methods.

Course Contents:

UNIT I
Introduction to finishing: Objective of finishing, classification, temporary, semipermanent and permanent finishing, their significance

UNIT II
Finishing equipments: Principles of functions of different machines used in finishing including padding units, ca142arbonizatinits, sanfosising machine, foam applicator

UNIT III
Different finishing processes: Objective, classification of different finishing processes, principle, methods, advantages and disadvantages of different finishing processes including cross linking agents used for different substrates to impart crease recovery/easy care finish, anti shrink finishing, flame retarding/proofing, water repelling, rot and mildew proofing of wool, ca142arbonizationf polyester/ cellulosic materials, application of softeners, enzymatic softening, organdie finish, milling of wool, moth proofing of wool, anti static finish, soil release finish, finishing of blends.

UNIT IV
Foam Application Technology: Concept objective, range of application, principle, ingredients of a foam system, half life of foam, blow ratio, foam generator, foam applicator, merits and demerits.

UNIT V
Colour and Chemical constitution and its estimation: Concept of colour and brief idea about the relation between colour and chemical constitution, factors governing transmission, Beer's law, Lambert's law, absorption, scattering of light, concept and definition of source, illuminant, detector, description of colour, colour co-ordinates, colour rendition, colour adaptation, hue, chroma, value, standard observer, tristimulus value, colour and colour difference measurement, Kubelka Munk's equation, methods of colour estimation, - manual, their limitations, instrumental, pass-fail criteria, concept of colour temperature, metamerism, standard illumination, absorption and reflectance spectra of a coloured material, extinction coefficient, measurement whiteness, evaluation of optical whitening, brief idea of computer colour matching and formulation.

UNIT VI
Evaluation of different fastness properties: Objectives of fastness measurement, different methods of fastness namely, wash, light, rubbing, perspiration, other allied fastness properties, their estimation methods.

Learning Resources:

Text Books

1. Handbook of Fiber Science and Technology, Vol. II, Chemical Processing of Fibers and Fabrics, Part A and B by M. Lewin and S.B. Sello
2. Principles of Colour Technology, F.W. Billmeyer
3. Dyeing and Chemical Technology of Textile Fibres by E.R. Trotman
4. An Introduction to Textile Finishing by J.T. Marsh
5. Textile Finishing by A.J. Hall
6. Colour Physics for Industry by R. McDonald
7. Chemical After-Treatment of Textiles by H. Mark, N.S. Wooding and S.M. Atlas
8. Instrumental Colour Measurement and Computer Aided Colour Matching for Textiles by H.S. Shah and R.S. Gandhi

References

1. Methods of Test for Colour Fastness of Textiles and Leather by The Society of Dyers and Colourists
2. ISI Handbook of Textile Testing by Bureau of Indian Standards
3. BS Handbook, Methods of Test for Textiles by British Standards Institution

Database management system

Course code: **L:T:P:3:0:0**

Rationale:

The course objectives are to introduce students to the fundamental concepts and principles of Database Management Systems (DBMS). It aims to provide a strong foundation in database design, data modeling, and query languages. The course also covers essential topics like transaction management, concurrency control, and database security.

Course Outcomes:

CO1: To understand the architecture and components of a Database Management System.
CO2: To apply Entity-Relationship (ER) modeling to design a relational database schema.
CO3: To write and execute queries using SQL for data definition, manipulation, and control.
CO4: To normalize a database schema to eliminate data redundancy and anomalies.
CO5: To understand transaction management, concurrency control, and database recovery techniques.

Course Contents:

UNIT – I

Database System Applications: A Historical Perspective, File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS
Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model

UNIT – II

Introduction to the Relational Model: Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical database design, introduction to views, destroying/altering tables and views Relational Algebra, Tuple relational Calculus, Domain relational calculus

UNIT – III

SQL: QUERIES, CONSTRAINTS, TRIGGERS: form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases
Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multivalued dependencies, FOURTH normal form, FIFTH normal form

UNIT – IV

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions

UNIT – V

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree based Indexing, Comparison of File Organizations,

Indexes- Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure

Learning Resources:

Text Books:

1. Database System Concepts, Silberschatz, Korth, McGraw Hill, V Edition, 3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata McGraw Hill

References:

1. Database Systems Design, Implementation, and Management, Peter Rob & Carlos Coronel, 7th Edition
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD
5. Database Systems Using Oracle: A Simplified Guide to SQL and PL/SQL, Shah, PHI
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition

Practical

Lab-I
(Compiler Design Lab)

Course code: **L:T:P:0:0:2**

Rationale:

The objective of this lab is to provide students with practical, hands-on experience in implementing the theoretical concepts of compiler design. Through a series of programming assignments, students will learn to construct the core components of a compiler, including the lexical analyzer, parser, and code generator. The course emphasizes the translation of a high-level language into an executable form, thereby fostering a deep understanding of the entire compilation process.

Course Outcomes:

CO1: To understand and implement the different phases of a compiler, such as lexical analysis, parsing, and code generation.
CO2: To apply compiler-generating tools like Lex/Flex and Yacc/Bison to automate the generation of a scanner and parser.
CO3: To design and write a predictive parser for a simple programming language.
CO4: To translate a high-level language into an intermediate representation, like three-address code.
CO5: To generate target code (assembly language) for a given source code and apply code improvement techniques.

List of experiments:

UNIT **I**
Familiarity with compiled codes (assembly language) of RISC and CISC machines.

UNIT **II**
Writing a scanner.

UNIT **III**
Writing predictive parser for a small language.

UNIT **IV**
Small experiment with scanner (lex/flex) and parser (yacc/bison) generator (such as translation of regular expression to NFA or the construction of parse tree).

UNIT **V**
Writing scanner-parse specification for a small language.

UNIT **VI**
Translation of the language to an intermediate form (e.g., three-address code).

UNIT **VII**
Generation of target code (in assembly language).

UNIT **VIII**
Code improvement (optional).

Learning Resources:

Text Books

1. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, *Compilers: Principles, Techniques and Tools*, Addison-Wesley.
2. Michael L. Scott, *Programming Language Pragmatics*, Elsevier.

Reference Books

1. Andrew W. Appel, *Modern Compiler Implementation in C/Java*, Cambridge University Press.
2. Keith D. Cooper and Linda Torczon, *Engineering a Compiler*, Elsevier.
3. Allen I. Holob, *Compiler Design in C*, Prentice-Hall.
4. Steven S. Muchnik, *Advanced Compiler Design and Implementation*, Elsevier.
5. Randy Allen and Ken Kennedy, *Optimizing Compilers for Modern Architectures*, Elsevier.

Lab-II
(Machine Learning LAB)

Course code: **L:T:P: 0:0:2**

Rationale:

The course is designed to provide a hands-on introduction to fundamental machine learning algorithms and techniques. Through a series of experiments, students will gain practical experience in implementing, simulating, and evaluating various models, from linear regression and decision trees to more complex concepts like neural networks. This practical approach allows students to understand the strengths and weaknesses of different algorithms and how to apply them to real-world datasets using popular tools like Python or R.

Course Outcomes:

CO1: To implement data pre-processing and exploratory data analysis techniques on a given dataset.

CO2: To deploy and simulate a variety of supervised learning algorithms, including linear regression and classification models.

CO3: To implement and compare the performance of different clustering algorithms like k-NN and k-Means.

CO4: To simulate a neural network and a genetic algorithm to solve complex problems.

CO5: To select and apply the appropriate machine learning algorithm for a specific problem statement and evaluate the model's performance**.**

List of experiments:

1. Implement data pre-processing
2. Deploy Simple Linear Regression
3. Simulate Multiple Linear Regression
4. Implement Decision Tree 1Credits
5. Deploy Random forest classification
6. Simulate Naïve Bayes algorithm
7. Implement K-Nearest Neighbors (K-NN), k-Means
8. Deploy Support Vector Machine, Apriori algorithm
9. Simulate Artificial Neural Network
10. Implement the Genetic Algorithm code

Suggested Tools: Python/R/MATLAB

Lab-III
(Cyber Security LAB)

Course code: **L:T:P: 0:0:2**

Rationale:

The lab provides a practical foundation in cyber security by exposing students to both offensive and defensive techniques in a controlled environment. It is designed to reinforce theoretical knowledge by allowing students to implement various cryptographic algorithms, set up security measures like firewalls, and simulate common attacks such as SQL injection and buffer overflows. This hands-on experience is crucial for understanding vulnerabilities and developing a mindset for building secure and robust systems.

Course Outcomes:

CO1: To implement and test symmetric and asymmetric encryption algorithms like AES, DES, and RSA.
CO2: To configure a firewall and a basic Intrusion Detection System (IDS) to defend a network.
CO3: To demonstrate and mitigate web-based attacks, such as SQL injection and Cross-Site Scripting (XSS).
CO4: To identify and exploit common software vulnerabilities like buffer overflow in a controlled environment.
CO5: To analyze and understand the behavior of malware and other cyber threats to develop effective detection and prevention strategies.

List of experiments:

1. Implement and test simple symmetric encryption algorithms like AES and DES.
2. Implement RSA encryption to demonstrate the concept of public and private keys.
3. Set up and configure a basic firewall using tools like iptables on Linux.
4. Demonstrate DNS spoofing and DNS cache poisoning attacks.
5. Set up a proxy server and demonstrate how attackers can use proxies to hide their tracks.
6. Demonstrate basic antiforensics techniques like
   i. Deleting logs
   ii. Using steganography tools.
7. Perform SQL injection on a test website and then implement measures to prevent it.
8. Create a simple application vulnerable to buffer overflow and demonstrate how to exploit it.
9. Implement an XSS attack on a test web application and demonstrate ways to mitigate such attacks.
10. Analyze a simple computer virus in a controlled environment and discuss detection and prevention strategies.
11. Investigate the functioning of a rootkit and demonstrate techniques to detect it.
12. Set up a basic IDS like Snort and test its effectiveness in detecting different types of attacks.

Learning Resources:

Text Books:

1. James Graham, Richard Howard, Ryan Olson, “Cyber Security Essentials”, CRC Press, Taylor & Francis Group, 2011.

References:

1. Mayank Bhusan, Rajkumar Singh Rathore, Aatif Jamshed, “Fundamental Of Cyber Security (Principles, Theory and Practices)”, BPB Publications, 2018.