JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
KAKINADA – 533 003, Andhra Pradesh, India
B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)

(Applicable from the academic year 2023-24 and onwards)

B.Tech. III Year I Semester

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

B.Tech. III Year II Semester

S.No	Category	Title	L	T	P	C
1	Professional Core	Compiler Design	3	0	0	3
2	Professional Core	Artificial Intelligence	3	0	0	3
3	Professional Core	Cryptography & Network Security	3	0	0	3
4	Professional Elective-II	1. Software Testing Methodologies 2. Cloud Computing 3. DevOps 4. Internet of Things 5. A minimum of 12 week MOOC Swayam / NPTEL course recommended by the BoS	3	0	0	3
5	Professional Elective-III	1.Software Project Management 2.Mobile Adhoc & Sensor Networks 3.Big Data Analytics 4.Distributed Operating System 5.A minimum of 12 week MOOC Swayam/ NPTEL course recommended by the BoS	3	0	0	3
6	Open Elective – II		3	0	0	3
7	Professional Core	Artificial Intelligence Lab	0	0	3	1.5
8	Professional Core	Compiler Design & Network Security Lab	0	0	3	1.5
9	Skill Enhancement course	Soft skills / SWAYAM Plus - 21st Century Employability Skills	0	1	2	2
10	Audit Course	Technical Paper Writing & IPR	2	0	0	-
		Total	**20**	**1**	**08**	**23**
Mandatory Industry Internship / Mini Project of 08 weeks duration during summer vacation						
MC	Minor Course (Student may select from the same specialized minors pool)		3	0	3	4.5
MC	Minor Course A minimum of 12 week, 3 credit course through SWAYAM / NPTEL		3	0	0	3
HC	Honors Course (Student may select from the same specialized honors pool)		3	0	0	3
HC	Honors Course A minimum of 12 week, 3 credit course through SWAYAM / NPTEL		3	0	0	3

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

B.Tech. IV Year I Semester

S. No	Category	Title	L	T	P	C
1	Professional Core	Machine Learning	2	0	2	3
2	Management Course- II	Human Resources & Project Management	2	0	0	2
3	Professional Elective-IV	1. Software Architecture & Design Patterns 2. Cyber Security 3. Augmented Reality & Virtual Reality 4. Deep Learning 5. A minimum of 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
4	Professional Elective-V	1. Agile Methodologies 2. Generative AI 3. Computer Vision 4. Cyber Physical Systems 5. A minimum of 12 week MOOC Swayam / NPTEL course recommended by the BoS	3	0	0	3
5	Open Elective-III		3	0	0	3
6	Open Elective-IV		3	0	0	3
7	Skill Enhancement Course	Prompt Engineering/ SWAYAM Plus - Certificate program in Prompt Engineering and ChatGPT	0	1	2	2
8	Audit Course	Constitution of India	2	0	0	-
9	Internship	Evaluation of Industry Internship /Mini Project	-	-	-	2
Total			**18**	**2**	**02**	**21**
MC	Minor Course (Student may select from the same specialized minors pool)		3	0	3	4.5
HC	Honors Course (Student may select from the same honors pool)		3	0	0	3
HC	Honors Course through SWAYAM/NPTEL (minimum 12 week, 3 credit course)		3	0	0	3

B.Tech. IV Year II Semester

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

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

Open Electives, offered to other department students:

Open Elective I: Principles of Operating Systems / Computer Organization and Architecture

Open Elective II: Principles of Database Management Systems / Data Structures for Data Science

Open Elective III: Object Oriented Programming through Java / Cyber Security

Open Elective IV: Principles of Software Engineering / Quantum Science & Technology

Minor Engineering

Note:

1. *To obtain Minor Engineering, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.*
2. *During Minor Course selection, there should not be any overlapping with Regular/Major/OPEN Electives*

Minor in CSE

Any three of the following courses in offline mode.

1.	Principles of Software Engineering	3-0-0-3
2.	Principles of Database Management Systems	3-0-3-4.5 (Mandatory)
3.	Advanced Data Structures & Algorithm Analysis	3-0-3-4.5 (Mandatory)
4.	Principles of Operating Systems	3-0-0-3

Any two of the following a minimum of 12 Week 3 credit NPTEL MOOC Courses / Relevant courses

1. Artificial Intelligence: Knowledge Representation and Reasoning
2. Computer Networks and Internet Protocol
3. Machine Learning and Deep Learning - Fundamentals and Applications
4. Fundamentals of Object Oriented Programming
5. Discrete Mathematics for CS

Minor in Quantum Technologies

Mandatory Courses

Survey of Quantum Technologies and Application	3-0-0-3
Foundations of Quantum Technologies	3-0-0-3

Atleast one of the following is Mandatory

Basic Programming Lab	2-0-2-3
Basic Laboratory Course for Quantum Technologies	2-0-2-3

Atleast one of the following is Mandatory

Introduction to Quantum Computation	3-0-0-3
Introduction to Quantum Communication	3-0-0-3
Introduction to Quantum Sensing	3-0-0-3
Introduction to Quantum Materials	3-0-0-3

Optional / Additional Courses (at most two)

Engineering Foundations of Quantum Technologies	3-0-0-3
Solid State Physics for Quantum Technologies	3-0-0-3
Quantum Optics	3-0-0-3
Quantum Algorithms and Cryptography	3-0-0-3
Quantum Cyber Security	3-0-0-3
Quantum Machine Learning	3-0-0-3

COURSES OFFERED FOR HONORS DEGREE IN CSE

Note: 1. *To obtain Honor's degree, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.*

2. *During Honors Course selection, there should not be any overlapping with Regular/Major/OPEN Electives*

Honor in CSE

Any three of the following courses in offline mode:

1. Principles of Secure Coding	3-0-0-3
2. Recommender Systems	3-0-0-3
3. Blockchain Technology	3-0-0-3
4. Natural Language Processing	3-0-0-3
5. High Performance Computing	3-0-0-3

Any three of the following a minimum of 12 Week 3 credit NPTEL MOOC Courses / Relevant Courses:

1. Social Network Analysis	12 Week 3 Credit Course, MOOCS
2. Applied Linear Algebra in AI & ML	12 Week 3 Credit Course, MOOCS
3. Design & Implementation of Human-Computer Interfaces – NPTEL MOOCS	
4. Privacy and Security in Online Social Media	12 Week 3 Credit Course, MOOCS
5. Computer Vision	- 12 Week 3 Credit Course, MOOCS
6. Applied Time-Series Analysis	12 Week 3 Credit Course, MOOCS
7. Parallel Computer Architecture	12 Week 3 Credit Course, MOOCS
8. Reinforcement Learning	12 Week 3 Credit Course, MOOCS
9. GPU Architecture and Programming	12 Week 3 Credit Course, MOOCS
10. Computational Complexity	12 Week 3 Credit Course, MOOCS
11. Quantum Algorithms and Cryptography	12 Week 3 Credit Course, MOOCS
12. Unmanned Arial Systems & Robotics	12 Week 3 Credit Course, MOOCS

III Year I Semester	DATA WAREHOUSING AND DATA MINING	L	T	P	C
		3	0	0	3

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

Course Objectives: The main objective of the course is to

- Introduce basic concepts and techniques of data warehousing and data mining
- Examine the types of the data to be mined and apply pre-processing methods on raw data
- Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.

UNIT–I:

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

UNIT II:

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

UNIT–III:

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

UNIT–IV:

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

UNIT–V:

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

Text Books:

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

Reference Books:

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

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

Course Objectives: The main objective of the course is to

- Provide insight about networks, topologies, and the key concepts.
- Gain comprehensive knowledge about the layered communication architectures (OSI and TCP/IP) and its functionalities.
- Understand the principles, key protocols, design issues, and significance of each layers in ISO and TCP/IP.
- Know the basic concepts of network services and various network applications.

UNIT I:

Introduction: Network Types, LAN, MAN, WAN, Network Topologies Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, OSI Vs TCP/IP.

Physical Layer –Introduction to Guided Media- Twisted-pair cable, Coaxial cable and Fiber optic cable and introduction about unguided media.

UNIT II:

Data link layer: Design issues, **Framing**: fixed size framing, variable size framing, flow control, error control, error detection and correction codes, CRC, Checksum: idea, one’s complement internet checksum, services provided to Network Layer, **Elementary Data Link Layer protocols**: simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel.

Sliding window protocol: One bit, Go back N, Selective repeat-Stop and wait protocol, Data link layer in HDLC, Point to point protocol (PPP)

UNIT – III:

Media Access Control: Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, **Controlled Access:** Reservation, Polling, Token Passing, **Channelization:** frequency division multiple Access (FDMA), time division multiple access (TDMA), code division multiple access(CDMA).

Wired LANs: Ethernet, Ethernet Protocol, Standard Ethernet, Fast Ethernet (100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet.

UNIT – IV:

The Network Layer Design Issues: Store and Forward Packet Switching, Services Provided to the Transport layer, Implementation of Connectionless Service, Implementation of Connection Oriented Service, Comparison of Virtual Circuit and Datagram Networks,

Routing Algorithms: The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control-Traffic Aware Routing-Admission Control-Traffic Throttling-Load Shedding. Traffic Control Algorithm - Leaky bucket & Token bucket.

Internet Working: How networks differ, How networks can be connected, Tunneling, internetwork routing, Fragmentation, network layer in the internet.

UNIT –V:

The Transport Layer: The Transport Service, Elements of Transport Protocols, Congestion Control, The internet transport protocols: UDP, TCP

Introduction to Application Layer: Introduction, Client Server Programming, WWW and HTTP, FTP, e-mail, TELNET, Secure Shell, Domain Name System, SNMP.

Text Books:

1. Computer Networks, Andrew S Tanenbaum, Fifth Edition. Pearson Education/PHI
2. Data Communications and Networks, Behrouz A. Forouzan, Fifth Edition TMH.

References Books:

1. Data Communications and Networks- Achut S Godbole, Atul Kahate
2. Computer Networks, Mayank Dave, CENGAGE

III Year I Semester	FORMAL LANGUAGES AND AUTOMATA THEORY	L	T	P	C
		3	0	0	3

Course Objectives: The main objective of the course is to

- Learn fundamentals of Regular and Context Free Grammars and Languages
- Understand the relation between Regular Language and Finite Automata and machines, Contexts free Languages, PDA and TM
- Learn how to design Automata's and machines as Acceptors, Verifiers and Translators, PDA as acceptor and TM as Calculators

UNIT I

Finite Automata: Need of Automata theory, Central Concepts of Automata Theory, Automation, Finite Automation, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with Є-Transitions, Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applications and Limitation of Finite Automata.

UNIT II

Regular Expressions, Regular Sets, Identity Rules, Equivalence of two RE, Manipulations of REs, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence between RG and FA, Inter Conversion.

UNIT III

Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, Є-Productions and Unit Productions, Normal Forms-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma, Closure Properties, Applications of Context Free Grammars.

UNIT IV

Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata.

UNIT V

Turning Machine: Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Types of TMs, Church's Thesis, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.

Text Books:

1. Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3^{rd} Edition, Pearson, 2008
2. Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3^{rd} Edition, PHI, 2007

Reference Books:

1. Elements of Theory of Computation, Lewis H.P. & Papadimition C.H., Pearson /PHI
2. Theory of Computation, V. Kulkarni, Oxford University Press, 2013
3. Theory of Automata, Languages and Computation, Rajendra kumar, McGraw Hill, 2014

e-Resources:

1) https://nptel.ac.in/courses/106/104/106104028/

III Year I Semester	OBJECT ORIENTED ANALYSIS AND DESIGN	L	T	P	C
		2	1	0	3

Course Objectives: The main objective of the course is to

- Become familiar with all phases of OOAD.
- Master the main features of the UML.
- Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.
- Learn the Object design Principles and understand how to apply them towards Implementation.

UNIT I:

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

UNIT II:

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

UNIT III:

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

UNIT IV:

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

UNIT V:

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

Text Books:

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

Reference Books:

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, McGraw-Hill. .

4. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

III Year I Semester	DESIGN AND ANALYSIS OF ALGORITHMS	L	T	P	C
		3	0	0	3

Course Objectives:

The main objective of the course is to

- Understand, analyze and denote time complexities of algorithms
- Introduce the different algorithmic approaches for problem solving through numerous example problems
- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.
- Provide some theoretical grounding in terms of finding the lower bounds of algorithms and the NP-completeness

UNIT I:

Introduction: Algorithm- introduction, Algorithm characteristics, Pseudo code Conventions Recursive Algorithm, Performance Analysis, Space Complexity, Time Complexity, Probabilistic Complexity, Asymptotic Notation, Practical Complexities, Big and little- O notations.

UNIT II:

Divide and Conquer: General Method, Defective chessboard, Binary Search, finding the maximum and minimum, Merge sort, Quick sort. The Greedy Method: The general Method, knapsack problem, Job sequencing with deadlines, minimum-cost spanning Trees, Optimal Merge Patterns, Single Source Shortest Paths.

UNIT III:

Dynamic Programming: General Method, Travelling Sales Person Problem, 0/1 Knapsack Problem, Optimal Binary Search Tree, Matrix Chain Multiplication, All pairs-shortest paths Floyd-Warshall algorithm for shortest path, String Editing, Reliability Design,.

UNIT IV:

Backtracking: The General Method, The 8-Queens problem, sum of subsets, Graph coloring, Hamiltonian cycles, knapsack problem.

Branch and Bound: The Method, Least cost (LC) Search, The 15- Puzzle: an Example, Control Abstraction for LC-Search.

UNIT V:

Bounding, FIFO Branch and-Bound, LC Branch and Bound, 0/1 Knapsack Problem, Traveling Salesperson problem, Introduction to NP-Hard and NP- Completeness - Decision Trees - P, NP and NP – Complete Problems - Cook's theorem.

Text Books:

1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", 2nd Edition, Universities Press.
2. Introduction to Algorithms Thomas H. Cormen, PHI Learning

Reference Books:

1. Harsh Bhasin, "Algorithms Design & Analysis", Oxford University Press.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "The Design and Analysis of

Computer Algorithms", Pearson Education India, 1 January 2004.

3. S. Sridhar, "Design and Analysis of Algorithms", Oxford University Press.

III Year I Semester	MICROPROCESSORS & MICROCONTROLLERS	L	T	P	C
		3	0	0	3

Course Objectives: The main objectives of the course are to

- Introduce fundamental architectural concepts of microprocessors and microcontrollers.
- Introduce 16 bit and 32 bit microcontrollers, impart knowledge on addressing modes and instruction set of 8086 and 8051
- Introduce assembly language programming concepts and explain memory and I/O interfacing with 8086 and 8051

UNIT I:

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

UNIT II:

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

UNIT III:

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

UNIT IV:

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

UNIT V:

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

Textbooks:

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

Reference Books:

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

III Year I Semester	QUANTUM COMPUTING	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course is to introduce the fundamentals of quantum computing, the problem-solving approach using finite dimensional mathematics

UNIT - I

Introduction to Quantum Computing: Bits Vs Qubits, Classical Vs Quantum logical operations, Hilbert space, Probabilities and measurements, Superposition, Entanglement, No cloning theorem, density operators and other quantum mechanical principles.

UNIT - II

Qubit: Quantum unit of information and its physical implementations. Bloch Sphere concept, Quantum Gates and Circuits: single qubit gates, multiple qubit gates, Bell states, design ofbasicquantum circuits.

UNIT - III

Quantum Algorithms: Realization of classical computation on quantum computers and simulator. Classical and quantum complexity theory. Quantum Parallelism, Quantum amplitude amplification, Quantum Phase Estimation methods.

UNIT - IV

Quantum Algorithms: Deutsch’s and Deutsch’s-Jozsa algorithm, Grover’s Search Algorithm,Simon’s Algorithm, Berstein-Vazirani Algorithm, Shor’s Factorization Algorithm.

UNIT - V

Quantum Information & Cryptography: Comparison between classical and quantum information theory. Super dense coding protocol, Quantum teleportation, Quantum Cryptography,Quantum Key Distribution - BB84, Ekert 91Protocol.

Text Books:

1. Quantum Computation and Quantum Information, Nielsen M. A.,Cambridge
2. Programming Quantum Computers, Essential Algorithms and Code Samples, Eric R Johnson, NicHarrigan, Mercedes Ginemo, Segovia, Oreilly

Reference Books:

1. Quantum Computing for Computer Scientists, Noson S. Yanofsk, Mirco A. Mannucci
2. Principles of Quantum Computation and Information,Benenti G., Casati G. and Strini G., Vol.I: Basic Concepts, Vol II
3. Basic Tools and Special Topics, World Scientific. Pittenger A. O., An Introduction to Quantum Computing Algorithms

III Year I Semester	DATA MINING LAB	L	T	P	C
		0	0	3	1.5

Pre-requisites: Data Base Management Systems, Python Programming

Course Objectives: The main objective of the course is to

- Inculcate Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment
- Design a data warehouse or data mart to present information needed by management in a form that is usable
- Emphasize hands-on experience working with all real data sets.
- Test real data sets using popular data mining tools such as WEKA, Python Libraries
- Develop ability to design various algorithms based on data mining tools.

Software Requirements: WEKA Tool/Python/R-Tool/Oracle Data mining

List of Experiments:

1. Creation of a Data Warehouse.

- Build Data Warehouse/Data Mart (using open source tools like Microsoft-SSIS)
- Design multi-dimensional data models namely Star, Snowflake and Fact Constellation schemas for any one enterprise (ex. Banking, Insurance, Finance, Healthcare, manufacturing, Automobiles, sales etc).
- Write ETL scripts and implement using data warehouse tools.
- Perform Various OLAP operations such slice, dice, roll up, drill up and pivot

2. Explore machine learning tool “WEKA”

- Explore WEKA Data Mining/Machine Learning Toolkit.
- Downloading and/or installation of WEKA data mining toolkit.
- Understand the features of WEKA toolkit such as Explorer, Knowledge Flow interface, Experimenter, command-line interface.
- Navigate the options available in the WEKA (ex. Select attributes panel, Preprocess panel, Classify panel, Cluster panel, Associate panel and Visualize panel)
- Study the arff file format Explore the available data sets in WEKA. Load a data set (ex. Weather dataset, Iris dataset, etc.)
- Load each dataset and observe the following:
  1. List the attribute names and they types
  2. Number of records in each dataset
  3. Identify the class attribute (if any)
  4. Plot Histogram
  5. Determine the number of records for each class.
  6. Visualize the data in various dimensions

3. Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets

- Explore various options available in Weka for preprocessing data and apply Unsupervised filters like Discretization, Resample filter, etc. on each dataset
- Load weather. nominal, Iris, Glass datasets into Weka and run Apriori Algorithm with different support and confidence values.
- Study the rules generated. Apply different discretization filters on numerical attributes and run the Apriori association rule algorithm. Study the rules generated.

- Derive interesting insights and observe the effect of discretization in the rule generation process.

4. Demonstrate performing classification on data sets Weka
   - Load each dataset and run Id3, J48 classification algorithm. Study the classifier output. Compute entropy values, Kappa statistic.
   - Extract if-then rules from the decision tree generated by the classifier, Observe the confusion matrix.
   - Load each dataset and perform Naïve-bayes classification and k-Nearest Neighbour classification. Interpret the results obtained.
   - Plot RoC Curves
   - Compare classification results of ID3, J48, Naïve-Bayes and k-NN classifiers for each dataset, and deduce which classifier is performing best and poor for each dataset and justify.

5. Demonstrate performing clustering of data sets
   - Load each dataset into Weka and run simple k-means clustering algorithm with different values of k (number of desired clusters).
   - Study the clusters formed. Observe the sum of squared errors and centroids, and derive insights.
   - Explore other clustering techniques available in Weka.
   - Explore visualization features of Weka to visualize the clusters. Derive interesting insights and explain.

6. Demonstrate knowledge flow application on data sets into Weka
   - Develop a knowledge flow layout for finding strong association rules by using Apriori, FP Growth algorithms
   - Set up the knowledge flow to load an ARFF (batch mode) and perform a cross validation using J48 algorithm
   - Demonstrate plotting multiple ROC curves in the same plot window by using j48 and Random forest tree

7. Demonstrate ZeroR technique on Iris dataset (by using necessary preprocessing technique(s)) and share your observations
8. Write a program to calculate chi-square value using Python/R. Report your observation.
9. Write a program of Naive Bayesian classification using Python/R programming language.
10. Write a program to cluster your choice of data using simple k-means algorithm using JDK
11. Write a program of cluster analysis using simple k-means algorithm using Python/R programming language.
12. Write a program to compute/display dissimilarity matrix (for your own dataset containing at least four instances with two attributes) using Python
13. Visualize the datasets using matplotlib in python/R.(Histogram, Box plot, Bar chart, Pie chart etc.,)

III Year I Semester	COMPUTER NETWORKS LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:
The main objectives of the course are to learn basic concepts of computer networking and acquire practical notions of protocols with the emphasis on TCP/IP.

List of Experiments:

1. Study of Network devices in detail and connect the computers in Local Area Network.
2. Write a C Program to implement the data link layer farming methods such as i) Character stuffing ii) bit stuffing.
3. Write a C Program to implement data link layer farming method checksum.
4. Write a C Program for Hamming Code generation for error detection and correction.
5. Write a C Program to implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
6. Write a C Program to implement Sliding window protocol for Goback N.
7. Write a C Program to implement Sliding window protocol for Selective repeat.
8. Write a C Program to implement Stop and Wait Protocol.
9. Write a C Program for congestion control using leaky bucket algorithm
10. Write a C Program to implement Dijkstra‘s algorithm to compute the Shortest path through a graph.
11. Write a C Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).
12. Wireshark
    i. Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.
13. How to run Nmap scan
14. Operating System Detection using Nmap
15. Do the following using NS2 Simulator
    i. NS2 Simulator-Introduction
    ii. Simulate to Find the Number of Packets Dropped
    iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
    iv. Simulate to Find the Number of Packets Dropped due to Congestion
    v. Simulate to Compare Data Rate& Throughput.

III Year I Semester	FULL STACK DEVELOPMENT - 2	L	T	P	C
		0	1	2	2

Course Objectives:

The main objectives of the course are to

- Make use of router, template engine and authentication using sessions to develop application in ExpressJS.
- Build a single page application using RESTful APIs in ExpressJS
- Apply router and hooks in designing ReactJS application
- Make use of MongoDB queries to perform CRUD operations on document database

Experiments covering the Topics:

- ExpressJS – Routing, HTTP Methods, Middleware, Templating, Form Data
- ExpressJS – Cookies, Sessions, Authentication, Database, RESTful APIs
- ReactJS – Render HTML, JSX, Components – function & Class, Props and States, Styles, Respond to Events
- ReactJS – Conditional Rendering, Rendering Lists, React Forms, React Router, Updating the Screen
- ReactJS – Hooks, Sharing data between Components, Applications – To-do list and Quiz
- MongoDB – Installation, Configuration, CRUD operations, Databases, Collections and Records

Sample Experiments:

1. ExpressJS – Routing, HTTP Methods, Middleware.

a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.

b. Write a program to accept data, retrieve data and delete a specified resource using http methods.

c. Write a program to show the working of middleware.

2. ExpressJS – Templating, Form Data

a. Write a program using templating engine.

b. Write a program to work with form data.

3. ExpressJS – Cookies, Sessions, Authentication

a. Write a program for session management using cookies and sessions.

b. Write a program for user authentication.

4. ExpressJS – Database, RESTful APIs

a. Write a program to connect MongoDB database using Mangoose and perform CRUD operations.

b. Write a program to develop a single page application using RESTful APIs.

5. ReactJS – Render HTML, JSX, Components – function & Class

a. Write a program to render HTML to a web page.

b. Write a program for writing markup with JSX.

c. Write a program for creating and nesting components (function and class).

6. ReactJS – Props and States, Styles, Respond to Events

a. Write a program to work with props and states.

b. Write a program to add styles (CSS & Sass Styling) and display data.
c. Write a program for responding to events.

7. ReactJS – Conditional Rendering, Rendering Lists, React Forms
a. Write a program for conditional rendering.
b. Write a program for rendering lists.
c. Write a program for working with different form fields using react forms.

8. ReactJS – React Router, Updating the Screen
a. Write a program for routing to different pages using react router.
b. Write a program for updating the screen.

9. ReactJS – Hooks, Sharing data between Components
a. Write a program to understand the importance of using hooks.
b. Write a program for sharing data between components.

10. MongoDB – Installation, Configuration, CRUD operations
a. Install MongoDB and configure ATLAS
b. Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()

11. MongoDB – Databases, Collections and Records
a. Write MongoDB queries to Create and drop databases and collections.
b. Write MongoDB queries to work with records using find(), limit(), sort(), createIndex(), aggregate().

12. Augmented Programs: (Any 2 must be completed)
a. Design a to-do list application using NodeJS and ExpressJS.
b. Design a Quiz app using ReactJS.
c. Complete the MongoDB certification from MongoDB University website.

Text Books:

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.
2. Node.Js in Action, Mike Cantelon, Mark Harter, T.J. Holowaychuk, Nathan Rajlich,Manning Publications. (Chapters 1-11)
3. React Quickly, AzatMardan,Manning Publications(Chapters 1-8,12-14)

Web Links:

1. ExpressJS - https://www.tutorialspoint.com/expressjs
2. ReactJS - https://www.w3schools.com/REACT (and) https://react.dev/learn#
3. MongoDB - https://learn.mongodb.com/learning-paths/introduction-to-mongodb

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

Course Objectives: The main objectives of the course are to

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

List of Experiments:
Students need to implement the following experiments

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

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

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

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

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

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

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

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

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

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

Text Books:

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

III Year II Semester	COMPILER DESIGN	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to understand the basic concept of compiler design, and its different phases which will be helpful to construct new tools like LEX, YACC, etc.

UNIT I:

Lexical Analysis: Language Processors, Structure of a Compiler, Lexical Analysis, The Role of the Lexical Analyzer, Bootstrapping, Input Buffering, Specification of Tokens, Recognition of Tokens, Lexical Analyzer Generator-LEX, Finite Automata, Regular Expressions and Finite Automata, Design of a Lexical Analyzer Generator.

Syntax Analysis: The Role of the Parser, Context-Free Grammars, Derivations, Parse Trees, Ambiguity, Left Recursion, Left Factoring,

UNIT II:

Top Down Parsing: Pre Processing Steps of Top Down Parsing, Backtracking, Recursive Descent Parsing, LL (1) Grammars, Non-recursive Predictive Parsing, Error Recovery in Predictive Parsing.

Bottom Up Parsing: Introduction, Difference between LR and LL Parsers, Types of LR Parsers, Shift Reduce Parsing, SLR Parsers, Construction of SLR Parsing Tables, More Powerful LR Parses, Construction of CLR (1) and LALR Parsing Tables, Dangling Else Ambiguity, Error Recovery in LR Parsing, Handling Ambiguity Grammar with LR Parsers.

UNIT III:

Syntax Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax Directed Translation, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's. **Intermediate Code Generation:** Variants of Syntax Trees, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Backpatching, Intermediate Code for Procedures.

UNIT IV:

Code Optimization: The Principle Sources of Optimization, Basic Blocks, Optimization of Basic Blocks, Structure Preserving Transformations, Flow Graphs, Loop Optimization, Data-Flow Analysis, Peephole Optimization

UNIT V:

Run Time Environments: Storage Organization, Run Time Storage Allocation, Activation Records, Procedure Calls, Displays

Code Generation: Issues in the Design of a Code Generator, Object Code Forms, Code Generation Algorithm, Register Allocation and Assignment.

Text Books:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson, 2007.

Reference Books:

1. Compiler Construction, Principles and Practice, Kenneth C Louden, Cengage Learning, 2006
2. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.

3. Optimizing Compilers for Modern Architectures, Randy Allen, Ken Kennedy, Morgan Kauffmann, 2001.
4. Levine, J.R., T. Mason and D. Brown, Lex and Yacc, edition, O'Reilly & Associates, 1990

III Year II Semester	ARTIFICIAL INTELLIGENCE	L	T	P	C
		3	0	0	3

Pre-requisite:

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

Course Objectives:

The student should be made to study the concepts of Artificial Intelligence and the methods of solving problems using Artificial Intelligence, learn different knowledge representation techniques and to introduce the concepts of Expert Systems and to understand the applications of AI, namely game playing, theorem proving, and machine learning.

UNIT - I

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

UNIT - II

Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A* ,AO* Algorithms, Problem reduction, Game Playing-Adversarial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

UNIT - III

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

UNIT - IV

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

UNIT - V

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

Textbooks:

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

Reference Books:

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

Online Learning Resources:

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

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

Course Objectives:

The main objectives of this course are to explore the working principles and utilities of various cryptographic algorithms including secret key cryptography, hashes and message digests, public key algorithms, design issues and working principles of various authentication protocols and various secure communication standards including Kerberos, IPsec, and SSL/TLS.

UNIT I:

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

UNIT II:

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

UNIT III:

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

UNIT IV:

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

UNIT V:

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

Text Books:

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

Reference Books:

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

III Year II Semester	SOFTWARE TESTING METHODOLOGIES	L	T	P	C
		3	0	0	3

Course Objectives: The main objectives of the course are to

- Provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- Develop skills in software test automation and management using the latest tools.

UNIT - I

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

UNIT - II

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

UNIT - III

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

UNIT - IV

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

UNIT - V

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

Text Books:

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

Reference Books:

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

		L	T	P	C
III Year II Semester	**CLOUD COMPUTING**	**3**	**0**	**0**	**3**

Course Objectives: The main objectives of the course are to

- Explain the evolving utility computing model called cloud computing and introduce the various levels of services offered by cloud.
- Discuss the fundamentals of cloud enabling technologies such as distributed computing, service-oriented architecture and virtualization.
- Emphasize the security and other challenges in cloud computing.
- Introduce the advanced concepts such as containers, serverless computing and cloud-centric Internet of Things.

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

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

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

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

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

Text Books:

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

Reference Books:

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

III Year II Semester	DEVOPS	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of this course are to:

- Describe the agile relationship between development and IT operations.
- Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
- Implement automated system update and DevOps lifecycle.

UNIT-I

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

UNIT-II

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

UNIT-III

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

UNIT-IV

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

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

UNIT-V

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

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

Text Books:

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

Reference Books:

1. Len Bass, Ingo Weber, Liming Zhu. DevOps: A Software Architect's Perspective. Addison Wesley; ISBN-10
2. Gene Kim Je Humble, Patrick Debois, John Willis. The DevOps Handbook, 1st Edition, IT Revolution Press, 2016.
3. Verona, Joakim Practical DevOps, 1st Edition, Packt Publishing, 2016.
4. Joakim Verona. Practical Devops, Ingram short title; 2nd edition (2018). ISBN10: 1788392574
5. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications. ISBN: 9788126579952

III Year II Semester	INTERNET OF THINGS	L	T	P	C
		3	0	0	3

Course Objectives:
The main objective of the course is to introduce the application areas of IOT, revolution of Internet in Mobile Devices, Cloud & Sensor Networks, building blocks of Internet of Things and characteristics

UNIT I:
Predecessors of IoT: Introduction, Wireless Sensor Networks, Machine-to-Machine Communications, Cyber Physical Systems
Emergence of IoT: Introduction, Evolution of IoT, Enabling IoT and the Complex Interdependence of Technologies, IoT Networking Components, Addressing Strategies in IoT

UNIT II:
IoT Sensing and Actuation: Introduction, Sensors, Sensor Characteristics, Sensorial Deviations, Sensing Types, Sensing Considerations, Actuators, Actuator Types, Actuator Characteristics
IoT Processing Topologies and Types: Data Format, Importance of Processing in IoT, Processing Topologies, IoT Device Design and Selection Considerations, Processing Offloading.

UNIT III:
IoT Connectivity Technologies: Introduction, IEEE 802.15.4, Zigbee, Thread, ISA100.11A, Wireless HART, RFID, NFC,DASH7, Z-Wave, Weightless, Sigfox, LoRa, NB-IT, Wi-Fi, Bluetooth
IoT Communication Technologies: Introduction, Infrastructure Protocols, Discovery Protocols, Data Protocols, Identification Protocols, Device Management, Semantic Protocols.

UNIT IV:
IoT Interoperability: Introduction, Standards, Frameworks
Fog Computing and Its Applications: Introduction, View of Fog Computing Architecture, Fog Computing in IoT, Selected Applications of Fog Computing

UNIT V:
Paradigms, Challenges, and the Future: Introduction, Evolution of New IoT Paradigms, Challenges Associated with IoT, Emerging Pillars of IoT
IoT Case Studies: Agricultural IoT, Vehicular IoT

Text Books:

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

Reference Books:

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

III Year II Semester	SOFTWARE PROJECT MANAGEMENT	L	T	P	C
		3	0	0	3

Course Objectives:

The main objective of the course is to

- Describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
- Compare and differentiate organization structures and project structures
- Implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

UNIT-I:

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

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

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

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

UNIT-II:

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

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

UNIT- III:

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

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

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

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

UNIT- IV:

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

Process Automation: Automation Building blocks, The Project Environment.

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

UNIT-V:

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

Text Books:

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

Reference Books:

1. Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH
2. Software Project Management, Joel Henry, PEA
3. Software Project Management in practice, Pankaj Jalote, PEA, 2005,
4. Effective Software Project Management, Robert K.Wysocki, Wiley,2006.
5. Project Management in IT, Kathy Schwalbe, Cengage

III Year II Semester	MOBILE ADHOC & SENSOR NETWORKS	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to

- Describe the Architecture sensor networks for various application setups.
- Devise appropriate data dissemination protocols and model links cost.
- Introduce the fundamental concepts of wireless sensor networks and basic knowledge of the various protocols at various layers.
- Evaluate the performance of sensor networks and identify bottlenecks.

UNIT I:

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

UNIT II:

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

UNIT III:

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

UNIT IV:

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

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

Text Books:

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

Reference Books:

1. Wireless Sensor Networks: An Information Processing Approach, 1st edition, *Feng Zhao, Leonidas Guibas*, Elsevier Science imprint, Morgan Kauffman Publishers, 2005, rp2009
2. Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications, 1st edition, Subir Kumar Sarkar, et al., Auerbach Publications, Taylor & Francis Group, 2008
3. Ad hoc Networking, 1stedition,*CharlesE.Perkins*, Pearson Education, 2001
4. Wireless Ad hoc Networking, 1st edition, *Shih-Lin Wu*, *Yu-Chee Tseng*, Auerbach Publications, Taylor & Francis Group, 2007
5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010

III Year II Semester	BIG DATA ANALYTICS	L	T	P	C
		3	0	0	3

Course Objectives: This course is aimed at enabling the students to

- Provide an overview of an exciting growing field of big data analytics.
- Introduce the tools required to manage and analyze big data like Hadoop, NoSQL, Map Reduce, HIVE, Cassandra, Spark.
- Teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability.
- Optimize business decisions and create competitive advantage with Big Data analytics

UNIT I:

Big data, convergence of key trends, unstructured data, industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data, credit risk management, big data and algorithmic trading, big data and healthcare, big data in medicine, advertising and big data, big data technologies, introduction to Hadoop, open source technologies, cloud and big data, mobile business intelligence, Crowd sourcing analytics, inter and trans firewall analytics.

UNIT II:

Introduction to NoSQL, aggregate data models, aggregates, key-value and document data models, relationships, graph databases, schema less databases, materialized views, distribution models, sharding, master-slave replication, peer- peer replication, sharding and replication, consistency, relaxing consistency, version stamps, Working with Cassandra ,Table creation, loading and reading data.

UNIT III:

Data formats, analyzing data with Hadoop, scaling out, Architecture of Hadoop distributed file system (HDFS), fault tolerance ,with data replication, High availability, Data locality , Map Reduce Architecture, Process flow, Java interface, data flow, Hadoop I/O, data integrity, compression, serialization. Introduction to Hive, data types and file formats, HiveQL data definition, HiveQL data manipulation, Logical joins, Window functions, Optimization, Table partitioning, Bucketing, Indexing, Join strategies.

UNIT IV:

Apache spark- Advantages over Hadoop, lazy evaluation, In memory processing, DAG, Spark context, Spark Session, RDD, Transformations- Narrow and Wide, Actions, Data frames ,RDD to Data frames, Catalyst optimizer, Data Frame Transformations, Working with Dates and Timestamps, Working with Nulls in Data, Working with Complex Types, Working with JSON, Grouping, Window Functions, Joins, Data Sources, Broadcast Variables, Accumulators, Deploying Spark- On-Premises Cluster Deployments, Cluster Managers- Standalone Mode, Spark on YARN , Spark Logs, The Spark UI- Spark UI History Server, Debugging and Spark First Aid

UNIT V:

Spark-Performance Tuning, Stream Processing Fundamentals, Event-Time and State full Processing - Event Time, State full Processing, Windows on Event Time- Tumbling Windows, Handling Late Data with Watermarks, Dropping Duplicates in a Stream, Structured Streaming Basics - Core Concepts, Structured Streaming in Action, Transformations on Streams, Input and Output.

Text Books:

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

Reference Books:

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

III Year II Semester	DISTRIBUTED OPERATING SYSTEM	L	T	P	C
		3	0	0	3

Course Objectives:
The main objective of the course is to introduce design issues and different message passing techniques in DOS, distributed systems, RPC implementation and its performance in DOS, distributed shared memory and resource management, distributed file systems and evaluate the performance in terms of fault tolerance, file replication as major factors

Unit I:
Fundamentals: Definition of Distributed Computing Systems. Evolution of Distributed Computing System; Distributed Computing System Models; Distributed Operating System, Issues in Designing a Distributed Operating System; Introduction to Distributed Computing Environment(DCE).
Message Passing: Introduction, Desirable features of a Good Message Passing System, Issues in PC by Message Passing, Synchronization, Buffering, Multi-datagram Messages, Encoding and Decoding of Message Data, Process Addressing, Failure Handling, Group Communication, Case Study: 4.3 BSD UNIX IPC Mechanism.

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

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

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

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

Text books
1. Pradeep. K. Sinha: Distributed Operating Systems: Concepts and Design, PHI, 2007.

Reference Books:
1. Andrew S. Tanenbaum: Distributed Operating Systems, Pearson Education, 2013.
2. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 2008
3. Sunita Mahajan, Seema Shan, “ Distributed Computing”, Oxford University Press,2015

III Year II Semester	ARTIFICIAL INTELLIGENCE LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

The main objective of the course is to provide students with hands-on experience and practical skills in AI technologies, fostering innovation and research, and enabling them to solve real-world problems using AI

List of Experiments:

1. Introduction of various Python libraries used for Artificial Intelligence.
2. Write a Python Program to implement Uninformed Search Technique: Breadth First Search
3. Write a Python Program to implement Uninformed Search Technique: Depth First Search
4. Write a Python Program to implement Informed Search Technique: A* Algorithm
5. Write a Python Program to implement Informed Search Technique: AO* Algorithm
6. Write a Python Program to implement Local Search Technique: Hill Climbing Algorithm
7. Write a Python Program to implement Game Playing Algorithms: Minimax and Alpha Beta Pruning
8. Create a Chatbot in Python
9. Write a Program to Implement N-Queens Problem using Python
10. Write a Program to Implement Missionaries Cannibals Problems using Python

III Year II Semester	COMPILER DESIGN & NETWORK SECURITY LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

The main objectives of the course is to learn basic understanding of compiler design, cryptography some key encryption techniques and implementing parsers, encryption and decryption using Ceaser Cipher, Substitution Cipher, Hill Cipher.

List of Experiments:

Compiler Design Lab:

1. Write a C program to identify different types of Tokens in a given Program.
2. Write a C program to Simulate Lexical Analyzer to validating a given input String.
3. Write a C program to implement the Brute force technique of Top down Parsing.
4. Write a C program to implement a Recursive Descent Parser.
5. Write C program to compute the First and Follow Sets for the given Grammar.
6. Write a C program for eliminating the left recursion and left factoring of a given grammar
7. Write a C program to check the validity of input string using Predictive Parser.
8. Write a C program for implementation of LR parsing algorithm to accept a given input string.

Network Security Lab:

1. Write a C program that contains a string (char pointer) with a value ‘Hello World’. The program should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value ‘Hello World’. The program should AND, OR and XOR each character in this string with 127 and display the result
3. Write a Java program to perform encryption and decryption using the following algorithms:
   - Ceaser Cipher
   - Substitution Cipher
   - Hill Cipher
4. Write a Java program to implement the DES algorithm logic
5. Write a JAVA program to implement the BlowFish algorithm logic
6. Write a JAVA program to implement the Rijndael algorithm logic.
7. Write a Java program to implement RSA Algorithm
8. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.

III Year II Semester	SOFT SKILLS	L	T	P	C
		0	1	2	2

Course Objectives: The main objectives of the course is to

- Equip the students with the skills to effectively communicate in English and train the students in interview skills, group discussions and presentation skills
- Motivate the students to develop confidence, enhance the students' interpersonal skills and improve the students' writing skills

UNIT – I

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

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

UNIT – II

Self-Management Skills: Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities

Etiquette: Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

UNIT – III

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

UNIT-IV

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

UNIT-V

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

Text books:

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

Reference books:

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

E-resources:

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

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

Course Objective: The course will explain the basic related to writing the technical reports and understanding the concepts related to formatting and structuring the report. This will help students to comprehend the concept of proofreading, proposals and practice

Unit I:
Introduction: An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.
Planning and Structuring: Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

Unit II:
Drafting report and design issues: The use of drafts, Illustrations and graphics.
Final edits: Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

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

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

Unit V:
Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of **Patenting and Development:** technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property

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

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

E-resources:

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

IV Year I Semester	MACHINE LEARNING	L	T	P	C
		2	0	2	3

Course Objectives:
The objectives of the course is to

- Define machine learning and its different types (supervised and unsupervised) and understand their applications.
- Apply supervised learning algorithms including decision trees and k-nearest neighbours (k-NN).
- Implement unsupervised learning techniques, such as K-means clustering.

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

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

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

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

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

List of Experiments:

Requirements: Develop the following program using Anaconda/Jupiter/Spider and evaluate ML models.

Experiment-1:
Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

Experiment-2:
For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

Experiment-3:
Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

Experiment-4:
Exercises to solve the real-world problems using the following machine learning methods: a) Linear Regression b) Logistic Regression c) Binary Classifier

Experiment-5: Develop a program for Bias, Variance, Remove duplicates , Cross Validation

Experiment-6: Write a program to implement Categorical Encoding, One-hot Encoding

Experiment-7:
Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.

Experiment-8:
Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions.

Experiment-9: Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Experiment-10:
Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

Experiment-11: Apply EM algorithm to cluster a Heart Disease Data Set. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

Experiment-12: Exploratory Data Analysis for Classification using Pandas or Matplotlib.

Experiment-13:
Write a Python program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set

Experiment-14:
Write a program to Implement Support Vector Machines and Principle Component Analysis

Experiment-15:
Write a program to Implement Principle Component Analysis

Text Books:
1. “Machine Learning Theory and Practice”, M N Murthy, V S Ananthanarayana, Universities Press (India), 2024

Reference Books:
1. “Machine Learning”, Tom M. Mitchell, McGraw-Hill Publication, 2017
2. “Machine Learning in Action”, Peter Harrington, Dream Tech
3. “Introduction to Data Mining”, Pang-Ning Tan, Michel Stenbach, Vipin Kumar, 7th Edition, 2019.

IV Year I Semester	HUMAN RESOURCES & PROJECT MANAGEMENT	L	T	P	C
		2	0	0	2

Course Objectives: The main objectives of the course are to

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

UNIT –I:

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

UNIT –II:

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

UNIT –III:

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

UNIT-IV:

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

UNIT-V:

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

Text Books:

1. Robert L. Mathis, John H. Jackson, Manas Ranjan Tripathy, Human Resource Management, Cengage Learning 2016.
2. Sharon Pande and Swapnalekha Basak, Human Resource Management, Text and Cases, Vikas Publishing, 2e, 2016.
3. Stewart R. Clegg, Torgeir Skyttermoen, Anne Live Vaagaasar, Project Management,

Sage Publications, 1e, 2021.

4. K. Nagarajan, Project Management, New Age International Publishers, 8e, 2017.

Reference Books :

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

IV Year I Semester	SOFTWARE ARCHITECTURE & DESIGN PATTERNS	L	T	P	C
		3	0	0	3

Course Objectives: The main objectives of the course are to

- Understand the basic concepts to identify state behavior of real world objects
- Apply Object Oriented Analysis and Design concepts to solve complex problems
- Construct various UML models using the appropriate notation for specific problem context
- Design models to Show the importance of systems analysis and design in solving complex problems using case studies
- Study of Pattern Oriented approach for real world problems

UNIT – I:

Introduction: Design pattern, Describing design patterns, the catalog of design pattern, organizing the catalog, design patterns solving design problems, selecting a design pattern, using a design pattern, object oriented development, key concepts of object oriented design other related concepts, benefits and drawbacks of the paradigm

UNIT – II:

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

UNIT – III:

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

UNIT – IV:

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

UNIT – V:

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

Text Books:

1. Object Oriented Analysis, Design and Implementation, Brahma Dathan, Sarnath Rammath , Universities Press, 2013
2. Design Patterns, Erich Gamma, Richard Helan, Ralph Johman, John Vlissides, PEARSON Publication, 2013

Reference Books:

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

IV Year I Semester	CYBER SECURITY	L	T	P	C
		3	0	0	3

Course Objectives:

The aim of the course is to

- Identify security risks and take preventive steps
- Understand the forensics fundamentals, evidence capturing process and preservation of digital evidence

UNIT I:

Introduction to Cybercrime: Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyberstalking, Cybercafe and Cybercrimes, Botnets. Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/CellPhones, Network and Computer Attacks.

UNIT II:

Tools and Methods : Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, Sniffers, Spoofing, Session Hijacking Buffer over flow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Foot Printing and Social Engineering, Port Scanning, Enumeration.

UNIT III:

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

UNIT IV:

Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.

UNIT V:

Cyber Crime Legal Perspectives: Introduction, Cybercrime and the Legal Landscape around the World, The Indian ITAct, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian ITAct, Amendments to the Indian ITAct, Cybercrime and Punishment, Cyberlaw, Technology and Students: Indian Scenario.

Text Books:

1. Sunit Belapure Nina Godbole "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", WILEY, 2011.
2. Nelson Phillips and Enfinger Steuart, "Computer Forensics and Investigations", Cengage Learning, New Delhi, 2009.

Reference Books:

1. Michael T. Simpson, Kent Backman and James E. Corley, "Hands on Ethical Hacking and Network Defence", Cengage, 2019.
2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
3. Alfred Basta, Nadine Basta, Mary Brown and Ravinder Kumar "Cyber Security and Cyber Laws" , Cengage,2018.

E-Resources:

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

IV Year I Semester	AUGMENTED REALITY & VIRTUAL REALITY	L	T	P	C
		3	0	0	3

Objectives: The objectives of the course are to

- Provide a foundation to the fast growing field of AR and make the students aware of the various AR concepts.
- Give historical and modern overviews and perspectives on virtual reality.

UNIT – I:

Introduction to Augmented Reality: Augmented Reality, Defining augmented reality, history of augmented reality, Examples, Related fields
Displays: Multimodal Displays, Visual Perception, Requirements and Characteristics, Spatial Display Model, Visual Displays
Tracking: Tracking, Calibration, and Registration, Coordinate Systems, Characteristics of Tracking Technology, Stationary Tracking Systems, Mobile Sensors

UNIT – II:

Computer Vision for Augmented Reality: Marker Tracking, Multiple-Camera Infrared Tracking, Natural Feature Tracking by Detection, Outdoor Tracking.
Interaction: Output Modalities, Input Modalities, Tangible Interfaces, Virtual User Interfaces on Real Surfaces, Augmented Paper, Multi-view Interfaces, Haptic Interaction
Software Architectures: AR Application Requirements, Software Engineering Requirements, Distributed Object Systems, Dataflow, Scene Graphs

UNIT – III:

Introduction to Virtual Reality: Defining Virtual Reality, History of VR, Human Physiology and Perception
The Geometry of Virtual Worlds: Geometric Models, Axis-Angle Representations of Rotation, Viewing Transformations
Light and Optics: Basic Behavior of Light, Lenses, Optical Aberrations, The Human Eye, Cameras, Displays

UNIT – IV:

The Physiology of Human Vision: From the Cornea to Photoreceptors, From Photoreceptors to the Visual Cortex, Eye Movements, Implications for VR
Visual Perception: Visual Perception - Perception of Depth, Perception of Motion, **Perception of Color Visual Rendering:** Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates, Immersive Photos and Videos

UNIT – V:

Motion in Real and Virtual Worlds: Velocities and Accelerations, The Vestibular System, Physics in the Virtual World, Mismatched Motion and Vection
Interaction: Motor Programs and Remapping, Locomotion, Social Interaction
Audio: The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering

Text Books:

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

Reference Books:

1. “AR Game Development”, Allan Fowler, 1st Edition, Apress Publications, 2018, ISBN 978-1484236178
2. “Understanding Virtual Reality: Interface, Application and Design”, William R Sherman and Alan B Craig, (The Morgan Kaufmann Series in Computer Graphics)”. Morgan Kaufmann Publishers, San Francisco, CA, 2002
3. “Developing Virtual Reality Applications: Foundations of Effective Design”, Alan B Craig, William R Sherman and Jeffrey D Will, Morgan Kaufmann, 2009
4. “Designing for Mixed Reality”, Kharis O'Connell, O'Reilly Media, Inc., 2016, ISBN:9781491962381
5. “Theory and applications of marker-based augmented reality”, Sanni Siltanen, Julkaisija, Utgivare Publisher. 2012. ISBN 978-951-38-7449-0
6. “Designing Virtual Systems: The Structured Approach”, Gerard Jounghyun Kim, 2005

IV Year I Semester	DEEP LEARNING	L	T	P	C
		3	0	0	3

Course Objectives: The main objective of the course is to make students:

- Understand the concept of CNN and transfer learning techniques, to apply it in the classification problems
- Use RNN for language modeling and time series prediction.
- Use auto encoder and deep generative models to solve problems with high dimensional data including text, image and speech

UNIT I:

Introduction and Overview: Course Overview and Motivation; Introduction to Image Formation, Capture and Representation; Linear Filtering, Correlation, Convolution.

Visual Features and Representations: Edge, Blobs, Corner Detection; Scale Space and Scale Selection; SIFT, SURF; HoG, LBP, etc.

Visual Matching: Bag-of-words, VLAD; RANSAC, Hough transform; Pyramid Matching; Optical Flow

UNIT II:

Deep Learning Review: Review of Deep Learning, Multi-layer Perceptrons, Backpropagation

Convolutional Neural Networks (CNNs): Introduction to CNNs; Evolution of CNN Architectures: AlexNet, ZFNet, VGG, InceptionNets, ResNets, DenseNets.

Visualization and Understanding CNNs: Visualization of Kernels; Backprop-to-image/Deconvolution Methods; Deep Dream, Hallucination, Neural Style Transfer; CAM,Grad-CAM, Grad-CAM++; Recent Methods (IG, Segment-IG, SmoothGrad)

UNIT III:

CNNs for Recognition, Verification, Detection, Segmentation: CNNs for Recognition and Verification (Siamese Networks, Triplet Loss, Contrastive Loss, Ranking Loss); CNNs for Detection: Background of Object Detection, R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD, RetinaNet; CNNs for Segmentation: FCN, SegNet, U-Net, Mask-RCNN

UNIT IV:

Recurrent Neural Networks(RNNs): Review of RNNs; CNN + RNN Models for Video Understanding: Spatio-temporal Models, Action/Activity Recognition

Attention Models: Introduction to Attention Models in Vision; Vision and Language: Image Captioning, Visual QA, Visual Dialog; Spatial Transformers; Transformer Networks

UNIT V:

Deep Generative Models: Review of (Popular) Deep Generative Models: GANs, VAEs; Other Generative Models: PixelRNNs, NADE, Normalizing Flows, etc

Recent Trends: Zero-shot, One-shot, Few-shot Learning; Self-supervised Learning; Reinforcement Learning in Vision; Other Recent Topics and Applications

Text Books:

1. Deep Learning- Ian Goodfellow, Yoshua Bengio and Aaron Courvile, MIT Press, 2017
2. Deep Learning with Python - Francois Chollet, Released December 2017, Publisher(s): Manning Publications, ISBN: 9781617294433
3. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence - Jon Krohn, Grant Beyleveld, AglaéBassens, Released September 2019, Publisher(s):

Addison-Wesley Professional, ISBN: 9780135116821

4. Deep Learning from Scratch - Seth Weidman, Released September 2019, Publisher(s): O'Reilly Media, Inc., ISBN: 9781492041412

Reference Books:

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

Web Link:

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

IV Year I Semester	AGILE METHODOLOGIES	L	T	P	C
		3	0	0	3

Course Objectives:
The main objectives of this course are to

- Introduce the important concepts of Agile software development Process
- Emphasize the role of stand-up meetings in software collaboration
- Impart the knowledge on values and principles in understanding agility

UNIT I :
Learning Agile: Agile, Getting Agile into your brain, Understanding Agile values, No Silver Bullet, Agile to the Rescue. A fractured perspective, The Agile Manifesto, Understanding the Elephant, Where to Start with a New Methodology.

UNIT II :
The Agile Principles: The 12 Principles of Agile Software, The Customer Is Always Right, Delivering the Project, Better Project Delivery for the Ebook Reader Project. Communicating and Working Together, Project Execution—Moving the Project Along, Constantly Improving the Project and the Team. The Agile Project: Bringing All the Principles Together

UNIT III :
SCRUM and Self-Organizing Teams: The Rules of Scrum, Act I: I Can Haz Scrum, Everyone on a Scrum Team owns the Project, Status Updates Are for Social Networks!, The Whole Team Uses the Daily Scrum, Feedback and the Visibility-Inspection-Adaptation Cycle, The Last Responsible Moment, Sprinting into a Wall, Sprints, Planning, and Retrospectives.
Scrum Planning And Collective Commitment: Not Quite Expecting the Unexpected, User Stories, Velocity, and Generally Accepted Scrum Practices, Victory Lap, Scrum Values Revisited.

UNIT IV :
XP And Embracing Change: Going into Overtime, The Primary Practices of XP, The Game Plan Changed, but We're Still Losing, The XP Values Help the Team Change Their Mindset, An Effective Mindset Starts with the XP Values, The Momentum Shifts, Understanding the XP Principles Helps You Embrace Change.
XP, Simplicity, and Incremental Design: Code and Design, Make Code and Design Decisions at the Last Responsible Moment, Final Score.

UNIT V:
Lean, Eliminating Waste, and Seeing the whole: Lean Thinking, Creating Heroes and Magical Thinking. Eliminate Waste, Gain a Deeper Understanding of the Product, Deliver As Fast As Possible.
Kanban, Flow, and Constantly Improving: The Principles of Kanban, Improving Your Process with Kanban, Measure and Manage Flow, Emergent Behavior with Kanban.
The Agile Coach: Coaches Understand Why People Don't Always Want to Change. The Principles of Coaching.

Text Books :

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

Reference Books:

1. Andrew Stellman, Jennifer Green, Head first Agile, O'Reilly, 2017.
2. Rubin K , Essential Scrum : A Practical Guide To The Most Popular Agile Process, Addison-Wesley, 2013

IV Year I Semester	GENERATIVE AI	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of the course are to introduce the basics of Generative AI, Text Generation, the process of generating videos and GAN & variants.

UNIT I :

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

UNIT II:

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

UNIT III:

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

UNIT IV:

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

UNIT V:

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

Text Books:

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

Reference Books:

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

IV Year I Semester	COMPUTER VISION	L	T	P	C
		3	0	0	3

Course Objectives:
The objectives of the course are to introduce the Fundamental Concepts related to sources, shadows and shading and the Geometry of Multiple Views.

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

UNIT-II:
Linear Filters: Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates, Edge Detection: Noise, Estimating Derivatives, Detecting Edges Texture0: Representing Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape from Texture.

UNIT-III:
The Geometry of Multiple Views: Two Views Stereopsis: Reconstruction, Human Stereopsis, Binocular Fusion, Using More Cameras Segmentation by Clustering: What Is Segmentation? Human Vision: Grouping and Getstalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

UNIT-IV:
Segmentation by Fitting a Model: The Hough Transform, Fitting Lines, Fitting Curves, Fitting as a Probabilistic Inference Problem, Robustness Segmentation and Fitting Using Probabilistic Methods: Missing Data Problems, Fitting, and Segmentation, The EM Algorithm in Practice, Tracking With Linear Dynamic Models: Tracking as an Abstract Inference Problem, Linear Dynamic Models, Kalman Filtering, Data Association, Applications and Examples

UNIT- V:
Geometric Camera Models: Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection Equations Geometric Camera Calibration: Least-Squares Parameter Estimation, A Linear Approach to Camera Calibration, Taking Radial Distortion into Account, Analytical Photogrammetry,
Case study: Mobile Robot Localization Model- Based Vision: Initial Assumptions, Obtaining Hypotheses by Pose Consistency, Obtaining Hypotheses by pose Clustering, Obtaining Hypotheses Using Invariants, Verification, Case study: Registration in Medical Imaging Systems, Curved Surfaces and Alignment.

Text Books:

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

Reference Books:

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

IV Year I Semester	CYBER PHYSICAL SYSTEMS	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course are to

- Introduce the core principles behind Cyber Physical Systems,
- Identify Security mechanisms of Cyber physical system,
- Describe Synchronization in Distributed Cyber-Physical Systems

UNIT I:

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

UNIT II:

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

UNIT III:

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

UNIT IV:

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

UNIT V:

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

Text Books:

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

Reference Books:

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

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

Course Objectives:

The main objectives of the course are to

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

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

Lab Experiments:

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

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

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

Lab Experiments:

1. Few-Shot vs. Zero-Shot Comparison: Design and execute a zero-shot prompt and a few-shot prompt (with 2–3 exemplar input-output pairs) for a chosen text task (e.g., sentiment classification or translation); compare outputs for accuracy, consistency, and adherence to examples.
2. Role-Based & Negative Prompting: Craft a role-based prompt to establish a specific persona (e.g., “You are a financial advisor…”); then create a negative prompt to suppress undesired content (e.g., “Do not mention any brand names”); evaluate how each influences the model’s response.

3. Constraint Specification & Iterative Refinement: Select an open-ended task (e.g., summarizing a technical article); issue a basic prompt; identify failures in length or format; refine the prompt by adding explicit constraints (word count, bullet format, etc.); document improvements over two refinement cycles.

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

Lab Experiments:

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

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

Lab Experiments:

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

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

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

Lab Experiments:

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

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

Course Objectives: The objectives of the course are to

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

UNIT–I:

History of Making of the Indian Constitution: History, Drafting Committee, (Composition & Working)

Philosophy of the Indian Constitution- Preamble, Salient, Features

UNIT–II:

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

UNIT–III:

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

UNIT–IV:

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

UNIT–V:

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

Text Books:

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

Reference Books:

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

Open Electives

Offered to other department students

III Year I Semester	PRINCIPLES OF OPERATING SYSTEMS (Open Elective I)	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course is to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

UNIT - I

Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems

System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging

UNIT - II

Processes: Process Concept, Process scheduling, Operations on processes, Inter-process communication.

Threads and Concurrency: Multithreading models, Thread libraries, Threading issues.

CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms.

UNIT – III

Synchronization Tools: The Critical Section Problem, Peterson’s Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.

Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks.

UNIT - IV

Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.

Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing

Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

UNIT - V

File System: File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management.

Text Books:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson , 2016

Reference Books:

1. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition, McGraw- Hill, 2013

Online Learning Resources:

1. https://nptel.ac.in/courses/106/106/106106144/
2. http://peterindia.net/OperatingSystems.html

III Year I Semester	COMPUTER ORGANIZATION AND ARCHITECTURE (Open Elective I)	L	T	P	C
		3	0	0	3

Course Objectives:
The purpose of the course is to introduce principles of computer organization and the basic architectural concepts. It provides an in depth understanding of basic organization, design, programming of a simple digital computer, computer arithmetic, instruction set design, micro programmed control unit, pipelining and vector processing, memory organization and I/O systems

UNIT I:
Basic Structure Of Computers: Computer Types, Functional unit, Basic Operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers.
Logic gates: Digital Logic gates,Two-level realizations using gates - AND-OR, OR-AND, NAND-NAND and NOR-NOR

UNIT II:
Sequential circuits I: Classification of sequential circuits (synchronous and asynchronous): basic flip-flops, truth tables and excitation tables (NAND RS latch, NOR RS latch, RS flip-flop. JK flip-flop, T flip-flop, D flip-flop with reset and clear terminals).Conversion of flip-flop to flip-flop, Race around condition, Master J-K flipflop
Register Transfer Language And Micro-operations: Register Transfer language. Register Transfer Bus and memory transfers, Arithmetic Micro-operations, Logic micro operations, shift micro operations, Arithmetic logic shift unit. Instruction codes. Computer Registers, Computer instructions, Instruction cycle.

UNIT III:
Micro Programmed Control: Control memory, Address sequencing, micro program example, design of control unit.
Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

UNIT IV:
Microprocessors: Evaluation of Microprocessors, CISC and RISC, Characteristics of Microprocessors
Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory, Cache memories performance considerations, Virtual memories Introduction to Shift registers and RAID

UNIT V:
Input – Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, DMA, Input Output Processor, Serial Communication.

Text Books:

1. Digital Logic and Computer Design, Moriss Mano, 11th Edition, Pearson Education.
2. Computer Organization, 5thed.,Hamacher, Vranesicand Zaky,TMH,2002
3. Computer System Architecture, 3/e, MorisMano, Pearson/PHI.

Reference Books:

1. Computer System Organization &Architecture, John D.Carpinelli, Pearson,2008
2. Computer System Organization, NareshJotwani, TMH, 2009
3. Computer Organization & Architecture: Designing for Performance, 7th ed., William Stallings, PHI, 2006
4. Structured Computer Organization, Andrew S. Tanenbaum, 4th Edition, PHI/Pearson.

III Year II Semester	PRINCIPLES OF DATABASE MANAGEMENT SYSTEMS (Open Elective II)	L	T	P	C
		3	0	0	3

Course Objectives:
The main objectives of the course is to

- Introduce database management systems and to give a good formal foundation on the relational model of data
- Introduce the concepts of SQL
- Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization
- Provide an overview of physical design of a database system, by discussing database storage techniques

UNIT I:
Introduction: Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database. [Text Book -2]

UNIT II:
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, Extended ER features [Text Book -1]

UNIT III:
Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/Altering Tables and Views [Text Book -1]]

UNIT IV:
BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).
SQL: Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations. [Text Book -1]

UNIT V:
Normal Forms: Introduction to Schema Refinement, Functional Dependencies, Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization. [Text Book -1]

Text Books:

1) Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4,5)
2) Database System Concepts,6th edition, Silberschatz, Korth, Sudarsan, TMH (For Chapter 1 and Chapter 5)

Reference Books:

1) Introduction to Database Systems, 8thedition, C J Date, Pearson.
2) Database Management System, 6th edition, Ramez Elmasri, Shamkant B. Navathe, Pearson
3) Database Principles Fundamentals of Design Implementation and Management, 10th edition, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022

Web-Resources:

1) https://nptel.ac.in/courses/106/105/106105175/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview

III Year II Semester	DATA STRUCTURES FOR DATA SCIENCE (Open Elective II)	L	T	P	C
		3	0	0	3

Course Objectives: The main objectives of the course are to

- Introduce fundamental data structures using Python in the context of data science.
- Develop problem-solving skills through structured data organization.
- Explore real-world data scenarios using appropriate data structures.
- Demonstrate how data structures influence the performance of data processing tasks.
- Prepare students for data analysis using structured and unstructured data.

UNIT I:

Introduction to Data Structures and Python Refresher: Python refresher: variables, data types, control structures, functions; Importance of data structures in data science; Introduction to complexity: time and space; Built-in Python data structures: lists, tuples, dictionaries, sets; Case Study: Handling simple datasets (CSV files) using lists and dictionaries.

UNIT II:

Linear Data Structures: Arrays vs Lists in Python; Stacks and Queues: concepts, operations, and use cases. Linked Lists: singly and doubly linked lists. Applications in parsing data streams and data preprocessing. Practical: Simulating real-world queues (e.g., customer service, job scheduling).

UNIT III:

Non-Linear Data Structures: Trees: binary trees, binary search trees, introduction to decision trees. Graphs: representation using adjacency list/matrix, traversal (BFS, DFS). Use in social networks, recommendation systems, clustering; Visualization with libraries: networkx, matplotlib.

UNIT IV:

Searching, Sorting, and Hashing: Linear and Binary Search. Sorting algorithms: bubble, selection, insertion, merge, quick (overview and applications). Hashing: concepts, hash functions, collisions, and resolution strategies; Applications: fast lookups in datasets, data indexing.

UNIT V:

Data Structures for Data Science Applications: Working with tabular data using Pandas DataFrame. Hierarchical data: JSON, XML – parsing and storing using trees/dictionaries. Time series data and sliding window techniques. Case Studies: Analyzing sensor data from IoT devices. Real-world data wrangling using Pandas and Numpy. Mini project: Data structure-based solution for a small data science problem.

Text Books :

1. "Problem Solving with Algorithms and Data Structures using Python" – Brad Miller and David Ranum
2. "Data Structures and Algorithms in Python" – Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser

Reference Books:

1. "Python for Data Analysis" – Wes McKinney

Online resources:

GeeksforGeeks, LeetCode, Kaggle datasets

IV Year I Semester	OBJECT ORIENTED PROGRAMMING THROUGH JAVA (Open Elective III)	L	T	P	C
		3	0	0	3

Course Objectives:

The learning objectives of this course are to:

- Identify Java language components and how they work together in applications
- Introduce the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- Introduce how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- Describe how to design applications with threads and how to use Java APIs for program development

UNIT I:

Object Oriented Programming: Basic concepts, Principles,

Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

Data Types, Variables, and Operators :Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator (=), Basic Arithmetic Operators, Increment (++) and Decrement (--) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

Control Statements: Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator?:, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.

UNIT II:

Classes and Objects: Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static

UNIT III:

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three dimensional Arrays, Arrays as Vectors.

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance.

Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

UNIT IV:

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Autoboxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class.

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

Java I/O and File: Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)

UNIT V:

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer.

Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

Java Database Connectivity: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, Result Set Interface

Text Books:

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object Oriented Programming, Debasis Samanta, Monalisa Sarma, Cambridge, 2023.
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

References Books:

1. The complete Reference Java, 11thedition, Herbert Schildt,TMH
2. Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

Online Resources:

1) https://nptel.ac.in/courses/106/105/106105191/
2)https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

IV Year I Semester	CYBER SECURITY (Open Elective III)	L	T	P	C
		3	0	0	3

Course Objectives:

The aim of the course is to

- Identify security risks and take preventive steps
- Understand the forensics fundamentals, the evidence capturing process and the preservation of digital evidence

UNIT I:

Introduction to Cybercrime: Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyberstalking, Cybercafe and Cybercrimes, Botnets. Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/CellPhones, Network and Computer Attacks.

UNIT II:

Tools and Methods : Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, Sniffers, Spoofing, Session Hijacking Buffer over flow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Foot Printing and Social Engineering, Port Scanning, Enumeration.

UNIT III:

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

UNIT IV:

Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.

UNIT V:

Cyber Crime Legal Perspectives: Introduction, Cybercrime and the Legal Landscape around the World, The Indian ITAct, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian ITAct, Amendments to the Indian ITAct, Cybercrime and Punishment, Cyberlaw, Technology and Students: Indian Scenario.

Text Books:

1. Sunit Belapure Nina Godbole “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, WILEY, 2011.
2. Nelson Phillips and Enfinger Steuart, “Computer Forensics and Investigations”, Cengage Learning, New Delhi, 2009.

Reference Books:

1. Michael T. Simpson, Kent Backman and James E. Corley, “Hands on Ethical Hacking and Network Defence”, Cengage, 2019.
2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
3. Alfred Basta, Nadine Basta, Mary Brown and Ravinder Kumar “Cyber Security and Cyber Laws” , Cengage,2018.

E-Resources:

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

IV Year I Semester	PRINCIPLES OF SOFTWARE ENGINEERING (Open Elective IV)	L	T	P	C
		3	0	0	3

Course Objectives:
The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

UNIT I:
Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.
Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT II:
Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, risk management.
Requirements Analysis and Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III:
Software Design: Overview of the design process, characterizing a good software design, Layered arrangement of modules, Cohesion and Coupling. approaches to software design.
Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)
Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.
User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT IV:
Coding and Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing**,** Testing object-oriented programs, Smoke testing, and Some general issues associated with testing.
Software Reliability and Quality Management: Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000. SEI Capability maturity model. Few other important quality standards, and Six Sigma**.**

UNIT V:
Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.
Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering: A Practitioner's Approach, Roger S. Pressman, 9th Edition, Mc-Graw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville,10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

e-Resources:

1) https://nptel.ac.in/courses/106/105/106105182/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

IV Year I Semester	**QUANTUM SCIENCE & TECHNOLOGY** **(Open Elective IV)**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Prerequisites: Basic Physics, Linear Algebra, and Introduction to Modern Physics

Course Objectives: The main objectives of the course are to

- Introduce fundamental concepts of quantum mechanics and its mathematical formalism.
- Explore quantum computing and communication principles and technologies.
- Understand the physical implementation and limitations of quantum systems.
- Enable students to relate quantum theory to practical applications in computing, cryptography, and sensing.
- Familiarize students with the emerging trends in quantum technologies.

Unit I:
Fundamentals of Quantum Mechanics: Historical background: Blackbody radiation, photoelectric effect, and Compton scattering; Dual nature of light and matter; De Broglie hypothesis; Schrödinger equation; Free particle, infinite potential well, step potential; Operators and observables: position, momentum, Hamiltonian;

Unit II:
Quantum Information Theory: Classical vs. quantum information; Qubit representation using Bloch sphere; Quantum superposition and quantum entanglement; Dirac notation (bra-ket), tensor products, and composite systems; Bell states and EPR paradox; Quantum gates: Pauli-X, Y, Z; Hadamard; Phase; T; CNOT; Quantum circuit models and notation;

Unit III: Quantum Computing: Classical computing review and limitations; Quantum parallelism and interference; Deutsch and Deutsch-Jozsa algorithms; Grover’s search algorithm, Oracle and amplitude amplification; Shor’s factoring algorithm (overview and significance);

Unit IV:
Quantum Fourier Transform (QFT); Quantum error correction: Bit-flip, phase-flip, and Shor’s 9-qubit code; Introduction to quantum programming: Qiskit, Cirq, IBM Quantum Experience (overview)

Unit V:
Quantum Technologies and Applications: Quantum sensors: magnetometry, gravimetry; Quantum metrology: standard time, atomic clocks; Quantum imaging and lithography; Hardware platforms: Superconducting qubits, Trapped ions, Photonic quantum processors; Quantum supremacy and NISQ era; Global initiatives: IBM, Google, D-Wave, IonQ, India’s NQM; Ethical concerns and future prospects

Text Books:

1. "Quantum Computation and Quantum Information", Michael A. Nielsen, Isaac L. Chuang
2. "Quantum Mechanics: Concepts and Applications", Nouredine Zettili

Minor
in
CSE

PRINCIPLES OF SOFTWARE ENGINEERING	L	T	P	C
	3	0	0	3

Course Objectives:
The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

UNIT I:
Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.
Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT II:
Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, risk management.
Requirements Analysis and Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III:
Software Design: Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design.
Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)
Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.
User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT IV:
Coding and Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing**,** Testing object-oriented programs, Smoke testing, and Some general issues associated with testing.
Software Reliability and Quality Management: Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000. SEI Capability maturity model. Few other important quality standards, and Six Sigma**.**

UNIT V:
Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.
Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering: A Practitioner's Approach, Roger S. Pressman, 9th Edition, Mc-Graw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville,10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

e-Resources:

1) https://nptel.ac.in/courses/106/105/106105182/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

PRINCIPLES OF DATABASE MANAGEMENT SYSTEMS	L	T	P	C
	3	0	3	4.5

Course Objectives:

The main objectives of the course is to

- Introduce database management systems and to give a good formal foundation on the relational model of data and the concepts of SQL
- Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization and provide an overview of physical design of a database system, by discussing database storage techniques
- Populate and query a database using SQL DDL/DML Commands and declare & enforce integrity constraints on a database
- Write Queries using advanced concepts of SQL and programming PL/SQL including procedures, functions, cursors and triggers

UNIT I:

Introduction: Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

UNIT II:

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance.

UNIT III:

BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).

SQL: Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.

UNIT IV:

Schema Refinement (Normalization): Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency Lossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form (BCNF).

UNIT V:

Transaction Concept: Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Text Books:

1) Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
2) Database System Concepts,5th edition, Silberschatz, Korth, Sudarsan, TMH (For Chapter 1 and Chapter 5)

Reference Books:

1) Introduction to Database Systems, 8thedition, C J Date, Pearson.
2) Database Management System, 6th edition, Ramez Elmasri, Shamkant B. Navathe, Pearson
3) Database Principles Fundamentals of Design Implementation and Management, 10th edition, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022

Web-Resources:

1) https://nptel.ac.in/courses/106/105/106105175/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview

Experiments covering the topics:

- DDL, DML, DCL commands
- Queries, nested queries, built-in functions,
- PL/SQL programming- control structures
- Procedures, Functions, Cursors, Triggers,

Sample Experiments:

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables), examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS,UNION, INTERSET, Constraints.
   Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5.
   i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
   ii. Insert data into student table and use COMMIT, ROLLBACK and
   SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.
8. Program development using creation of procedures, passing parameters IN and OUT of PROCEDURES.

9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

Text Books/Suggested Reading:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lans, “Introduction to SQL”, Fourth Edition, Pearson Education, 2007

	L	T	P	C
ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS	**3**	**0**	**3**	**4.5**

Course Objectives:

The main objectives of the course are to

- Provide knowledge on advance data structures frequently used in Computer Science domain
- Develop skills in algorithm design techniques popularly used
- Understand the use of various data structures in the algorithm design
- Acquire practical skills in constructing and managing Data structures
- Apply the popular algorithm design methods in problem-solving scenarios

UNIT – I:

Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations.
AVL Trees – Creation, Insertion, Deletion operations and Applications
B-Trees – Creation, Insertion, Deletion operations and Applications

UNIT – II:

Heap Trees (Priority Queues) – Min and Max Heaps, Operations and Applications
Graphs – Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications
Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen's matrix multiplication, Convex Hull

UNIT – III:

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths
Dynamic Programming: General Method, All pairs shortest paths, Single Source Shortest Paths– General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing, Travelling Salesperson problem

UNIT – IV:

Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem
Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson Problem

UNIT – V:

NP Hard and NP Complete Problems: Basic Concepts, Cook's theorem
NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP)
NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling

Textbooks:

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh, 2ndEdition Universities Press
2. Computer Algorithms in C++, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 2nd Edition University Press

Reference Books:

1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C & C++: Langsam, Augenstein&Tanenbaum, Pearson, 1995
5. Algorithms + Data Structures & Programs:, N.Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni& Mehta, Galgottia Pub.
7. Data structures in Java:, Thomas Standish, Pearson Education Asia

Online Learning Resources:

1. https://www.tutorialspoint.com/advanced_data_structures/index.asp
2. http://peterindia.net/Algorithms.html
3. Abdul Bari, Introduction to Algorithms (youtube.com)

Experiments covering the Topics:

- Operations on AVL trees, B-Trees, Heap Trees
- Graph Traversals
- Sorting techniques
- Minimum cost spanning trees
- Shortest path algorithms
- 0/1 Knapsack Problem
- Travelling Salesperson problem
- Optimal Binary Search Trees
- N-Queens Problem
- Job Sequencing

Sample Programs:

1. Construct an AVL tree for a given set of elements which are stored in a file. And implement insert and delete operation on the constructed tree. Write contents of tree into a new file using in-order.
2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array. Implement searching, insertion and deletion operations.
3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.
4. Implement BFT and DFT for given graph, when graph is represented by
   a) Adjacency Matrix b) Adjacency Lists
5. Write a program for finding the biconnected components in a given graph.
6. Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases).
7. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
8. Implement Job Sequencing with deadlines using Greedy strategy.
9. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
10. Implement N-Queens Problem Using Backtracking.
11. Use Backtracking strategy to solve 0/1 Knapsack problem.
12. Implement Travelling Sales Person problem using Branch and Bound approach.

Reference Books:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, Sahni Sartaj, Mehta, Dinesh, 2nd Edition, Universities Press

2. Computer Algorithms/C++ Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 2nd Edition, University Press
3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
4. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill

Online Learning Resources:

1. http://cse01-iiith.vlabs.ac.in/

PRINCIPLES OF OPERATING SYSTEMS	L	T	P	C
	3	0	0	3

Course Objectives:

The main objectives of the course is to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

UNIT – I:

Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems

System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging

UNIT – II:

Processes: Process Concept, Process scheduling, Operations on processes, Inter-process communication.

Threads and Concurrency: Multithreading models, Thread libraries, Threading issues.

CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

UNIT – III:

Synchronization Tools: The Critical Section Problem, Peterson’s Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.

Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks.

UNIT – IV:

Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.

Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing

Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

UNIT – V:

File System: File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management; File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing.

Text Books:

3. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
4. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson , 2016

Reference Books:

2. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition,

McGraw- Hill, 2013

Online Learning Resources:

3. https://nptel.ac.in/courses/106/106/106106144/
4. http://peterindia.net/OperatingSystems.html

Minor
in
Quantum Technologies

SURVEY OF QUANTUM TECHNOLOGIES AND APPLICATION	L	T	P	C
	3	0	0	3

Course Objectives:

This course is meant to give an overview of the field of quantum technologies and make the students familiar with the state-of-the-art in all four verticals. The emphasis is not on depth in this course, but on covering the exciting aspects of the field.

UNIT – 1:

Quantum Technologies – four verticals: Motivation for Quantum Technologies

A qualitative overview of salient aspects of quantum physics: Quantum States, Wave functions, Probabilistic interpretation, Physical observables, Hermitian operators, expectation values,

Heisenberg uncertainty principle, Schrodinger equation, Time evolution; distinction from classical physics; Heuristic description of Superposition, Tunnelling and entanglement; No cloning theorem; Simulating classical systems – Feynman's idea of a quantum simulator and the birth of the field.

UNIT-II:

Quantum Computation: Basics of qubits -- How is it different from a classical bit. – Review of classical logic gates; Di Vincenzo criteria for realizing qubits; Basics of qubit gates and quantum circuits; Physical implementation of qubits (very qualitative description);Solid State Qubits: Semiconducting Qubits – quantum dots, spins, Superconducting Qubits – charge, flux and phase, Topological Qubits – proposals and advantages;

Atoms and Ions: Trapped ions, Rydberg atoms, Neutral atoms; Photonic Qubits: Conventional linear optical setups, Integrated Photonics; NMR qubits: Conventional NMR qubits, NV centres

Overview of applications and recent achievements: RSA and Shor's algorithm, Quantum Advantage; Long term goals and strategies being followed :Error correction

UNIT-III:

Quantum Sensing: Basics of quantum sensing, Basics of Photon (single and entangled) generation and detection, Gravimetry, Atomic clock, Magnetometry, State of the art in Quantum Sensing

UNIT-IV:

Quantum Communications: Basics of digital communication, Quantifying classical information – Shannon entropy, Basic ideas of quantum communication, security, eavesdropping, Overview of quantum communication achievements :Terrestrial – fibre-based, Free space, Satellite-based

UNIT-V:

Introduction to Quantum Materials: What are quantum materials, Why are they important, Applications (quantum computing, spintronics, etc.)

Overview of Key Classes of Quantum Materials: Topological Insulators, Superconductors, Mott Insulators, 2D Materials and Quantum Spin Liquids.

Course References:

1. Quantum Information Science – Manenti R., Motta M., 1st Edition, Oxford University Press (2023)

2. Quantum computation and quantum information – Nielsen M. A., and Chuang I. L., 10th Anniversary edition, Cambridge University Press (2010)
3. Elements of Quantum Computation and Quantum Communication, A. Pathak, Boca Raton, CRC Press (2015)
4. An Introduction to Quantum Computing, Phillip Kaye, Raymond Laflamme, and Michele Mosca, Oxford University Press (2006)
5. Quantum computing explained, David McMahon, Wiley (2008)

FOUNDATIONS OF QUANTUM TECHNOLOGIES	L	T	P	C
	3	0	0	3

Course Objectives:

This course is meant for laying down the central theoretical aspects of quantum mechanics in a rigorous manner where students learn the techniques and develop a good intuition for quantum physics.

UNIT-I:

Brief overview of classical physics (This segment is meant for the student to understand what a Hamiltonian is, which will feature later in quantum mechanics) : Hamiltonian function and Hamilton's equations, Phase-space description of a system, Connection and Equivalence with Newton's laws for simple systems – free particle, particle moving in a conservative potential, examples of Harmonic oscillator, hydrogen atom

Historical evolution of quantum mechanics: Planck's quantum hypothesis, Photo electric effect, Atomic spectra, Bohr's quantisation principle, De Broglie's Wave particle duality

UNIT-II:

Postulates of Quantum Mechanics: State vectors and Hilbert Space, Dirac Bra-Ket notation, Measurables and Hermitian Operators, Unitary Transformations, Schrodinger Equation and Time evolution of quantum states, Measurement Postulate, Schrodinger, Heisenberg and Interaction pictures, Eigen values, Expectation values and Matrix elements, Heisenberg's Uncertainty principle

UNIT-III:

Density operator formalism of quantum mechanics – pure and mixed states; Superposition and Entanglement in quantum mechanics; No cloning theorem; Applications of postulates – Particle in a box, Hydrogen atom, Harmonic Oscillator

Number states, ladder operators and Coherent states of a harmonicoscillator; Spin and Angular momentum – spin half particles; Rabi problem of a spin-half particle in a rotating magnetic field; Bosons and Fermions

UNIT-IV:

Statistical Physics: Quick review of first and second laws of thermodynamics, Thermal Equilibrium and Gibbs principle, Applying Gibbs principle to Classical and Quantum harmonicoscillators, Bosons and Fermions and Quantum statistics – Fermi-Dirac and Bose-Einstein distributions

UNIT - V:

Information Science: Digital communication and information: Quantifying information in terms of Shannon entropy; Basic ideas of quantum information; Decoherence and noise; Introductory ideas of Kraus operators

Brief overview of Computational Complexity: Qualitative ideas of a Turing machine: Types of Turing machines; Time and Space complexity – P vs NP, PSPACE; Quantum complexity classes – Q, EQP, BQP, BPP, QMA; Post Quantum Cryptography (PQC)

Course References:

1. Introduction to Quantum Mechanics, Griffiths D. J., 3rd Edition, Cambridge University Press (2024)
2. Introduction to Electrodynamics, Griffiths D. J., 4th edition, Cambridge University Press(2020)

3.Principles of Quantum Mechanics, Shankar, R., 2nd edition, Springer (2014)
5.Quantum Information Science – Manenti R., Motta M., 1st Edition, Oxford University Press (2023)
6.Quantum computation and quantum information – Nielsen M. A., and Chuang I. L., 10thAnniversary edition, Cambridge University Press (2010)
7.A Pathak, Elements of Quantum Computation and Quantum Communication, Boca Raton, CRC Press (2015)
8.Information Theory, Robert B. Ash, Dover Publications (2003)
9.Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage India Pvt. Ltd.(2014)
10. Statistical Mechanics, Pathria R. K., Paul D. Beale, 4th edition, Academic Press, (2021)

The Syllabus of the following courses will be communicated soon.

- Basic Programming Lab
- Basic Laboratory Course For Quantum Technologies
- Introduction To Quantum Computation
- Introduction To Quantum Communication
- Introduction To Quantum Sensing
- Introduction To Quantum Materials
- Engineering Foundations Of Quantum Technologies
- Solid State Physics For Quantum Technologies
- Quantum Optics
- Quantum Cyber Security
- Quantum Machine Learning

Honor
in
CSE

PRINCIPLES OF SECURE CODING	L	T	P	C
	3	0	0	3

Course Objectives: The main objectives of the course are to

- Introduce various security attacks and knowledge to recognize and remove common coding errors that lead to vulnerabilities.
- Outline the techniques for developing a secure application.
- Recognize opportunities to apply secure coding principles.

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

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

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

UNIT–IV:
Database and Web Specific Input Issues- Quoting the Input, Use of stored procedures, Building SQL statements securely, XSS related attacks and remedies

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

Text Book:

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

Reference Books:

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

RECOMMENDER SYSTEMS	L	T	P	C
	3	0	0	3

Course Objectives:
This course covers the basic concepts of recommender systems, including personalization algorithms, evaluation tools, and user experiences

UNIT-I:
Introduction: Recommender system functions, Linear Algebra notation: Matrix addition, Multiplication, transposition, and inverses, covariance matrices, Understanding ratings, Applications of recommendation systems, Issues with recommender system.

UNIT-II:
Collaborative Filtering: User-based nearest neighbour recommendation, Item-based nearest neighbour recommendation, Model based and pre-processing based approaches, Attacks on collaborative recommender systems.

UNIT-III:
Content-based recommendation: High level architecture of content-based systems, Advantages and drawbacks of content based filtering, Item profiles, discovering features of documents, obtaining item features from tags, representing item profiles, Methods for learning user profiles, Similarity based retrieval, Classification algorithms.
Knowledge based recommendation: Knowledge representation and reasoning, Constraint based recommenders, Case based recommenders.

UNIT-IV:
Hybrid approaches: Opportunities for hybridization, Monolithic hybridization design: Feature combination, Feature augmentation, Parallelized hybridization design: Weighted, Switching, Mixed, Pipelined hybridization design: Cascade Meta-level, Limitations of hybridization strategies.

UNIT-V:
Evaluating Recommender System: Introduction, General properties of evaluation research, Evaluation designs, Evaluation on historical datasets, Error metrics, Decision-Support metrics, User-Centred metrics.
Recommender Systems and communities: Communities, collaboration and recommender systems in personalized web search, Social tagging recommender systems, Trust and recommendations

Text Books:

1. Jannach D., Zanker M. , FelFering A., Recommender Systems: An Introduction, Cambridge University Press(2011), 1st ed.
2. Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer(2011), 1st ed.

Reference Books:

1. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer (2013), 1st ed.

BLOCKCHAIN TECHNOLOGY	L	T	P	C
	3	0	0	3

Course Objectives: The objectives of the course are to

1. Describe the fundamentals of Block Chain and various types of block chain and consensus mechanism.
2. Understand public block chain system, Private block chain system and consortium block chain.
3. Introduce the security issues of blockchain technology.

UNIT – I:

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

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

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

UNIT – II:

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

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

UNIT – III:

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

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

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

UNIT – IV:

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

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

UNIT – V:

Blockchain Case Studies:
Case Study 1 – Retail,
Case Study 2 – Banking and Financial Services,

Case Study 3 – Healthcare,
Case Study 4 – Energy and Utilities.
Blockchain Platform using Python: Introduction, Learn How to Use Python Online Editor, Basic Programming Using Python, Python Packages for Blockchain.
Blockchain platform using Hyperledger Fabric: Introduction, Components of Hyperledger Fabric Network, Chain codes from Developer.ibm.com, Blockchain Application Using Fabric Java SDK.

Text Book:

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

Reference Books:

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

NATURAL LANGUAGE PROCESSING	L	T	P	C
	3	0	0	3

Course Objectives:
This course introduces the fundamental concepts and techniques of natural language processing (NLP), computational properties of natural languages and the commonly used algorithms for processing linguistic information.

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

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

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

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

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

Text Books:

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2ndEdition, Daniel Jurafsky, James H. Martin - Pearson Publication,2014.
2. Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, OReilly Media,2009.

Reference Books:

1. Language Processing with Java and Ling Pipe Cookbook, 1stEdition, Breck Baldwin, Atlantic Publisher, 2015.

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

HIGH PERFORMANCE COMPUTING	L	T	P	C
	3	0	0	3

Course Objectives:
The main objectives of the course is to study parallel computing hardware and programming models, performance analysis and modeling of parallel programs

Unit I:
Introduction: Motivating Parallelism, Scope of Parallel Computing, Parallel Programming Platforms: Implicit Parallelism, Trends in Microprocessor and Architectures, Limitations of Memory, System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines, Scalable design principles, Architectures: N-wide superscalar architectures, Multi-core architecture.

Unit II:
Parallel Programming : Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models, The Age of Parallel Processing, the Rise of GPU Computing, A Brief History of GPUs, Early GPU.

Unit III:
Basic Communication: Operations- One-to-All Broadcast and All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All-to-All Personalized Communication, Circular Shift, Improving the Speed of Some Communication Operations. Programming shared address space platforms: threads- basics, synchronization, OpenMP programming

Unit IV:
Analytical Models: Sources of overhead in Parallel Programs, Performance Metrics for Parallel Systems, and The effect of Granularity on Performance, Scalability of Parallel Systems, Minimum execution time and minimum cost, optimal execution time. Dense Matrix Algorithms: MatrixVectorMultiplication, Matrix-Matrix Multiplication.

Unit V:
Parallel Algorithms- Sorting and Graph : Issues in Sorting on Parallel Computers, Bubble Sort and its Variants, Parallelizing Quick sort, All-Pairs Shortest Paths, Algorithm for sparse graph, Parallel Depth-First Search, Parallel BestFirst Search.
CUDA Architecture :CUDA Architecture, Using the CUDA Architecture, Applications of CUDA Introduction to CUDA C-Write and launch CUDA C kernels, Manage GPU memory, Manage communication and synchronization, Parallel programming in CUDA- C.

Text Books:

1. AnanthGrama, Anshul Gupta, George Karypis, and Vipin Kumar, "Introduction to Parallel Computing", 2nd edition, Addison-Wesley, 2003, ISBN: 0-201-64865-2
2. Jason sanders, Edward Kandrot, “CUDA by Example”, Addison-Wesley, ISBN-13: 978-0-13-138768-3

Reference Books :

1. Kai Hwang, ”Scalable Parallel Computing”, McGraw Hill 1998, ISBN:0070317984
2. Shane Cook, “CUDA Programming: A Developer's Guide to Parallel Computing

with GPUs", Morgan Kaufmann Publishers Inc. San Francisco, CA, USA 2013 ISBN: 9780124159884

3. David Culler Jaswinder Pal Singh, "Parallel Computer Architecture: A Hardware/Software Approach", Morgan Kaufmann,1999, ISBN 978-1-55860-343-1
4. Rod Stephens, "Essential Algorithms", Wiley, ISBN: 978-1-118-61210-1