MATA TRIPURA SUNDARI OPEN UNIVERSITY, TRIPURA

Registrar
Mata Tripura Sundari Open University
Gomati-Tripura

PROGRAMME PROJECT REPORT
MASTER OF COMPUTER APPLICATION (MCA)
2025–26

Introduction

The Master of Computer Applications (MCA) programme stands at the intersection of technology, innovation, and problem-solving, offering a comprehensive educational journey for individuals aspiring to excel in the dynamic field of computer science and information technology. With its blend of theoretical knowledge, practical skills, and real-world application, the MCA programme serves as a gateway to a myriad of career opportunities in today's digital age. At its core, the MCA program is designed to provide students with a deep understanding of fundamental computer science principles and their practical applications. Through a carefully crafted curriculum, students explore a wide range of topics including software development, database management, computer networks, cyber security, artificial intelligence, and emerging technologies. By mastering these concepts, students develop the critical thinking skills and technical expertise necessary to tackle complex challenges in the rapidly evolving world of technology.

One of the distinguishing features of the MCA programme is its emphasis on hands-on learning and practical experience. Beyond the classroom, students engage in experiential learning opportunities such as internships, industry projects, and research collaborations, allowing them to apply their knowledge in real-world settings and gain valuable insights into industry best practices. These experiences not only enhance students' skill sets but also provide them with a competitive edge in the job market upon graduation.

Furthermore, the MCA programme fosters a culture of innovation and entrepreneurship, encouraging students to explore their creativity and pursue innovative solutions to real-world problems. Whether through student-led projects, hackathons, or entrepreneurship initiatives, students are empowered to unleash their potential and make a positive impact in the world through technology. In addition to academic excellence, the MCA programme is committed to nurturing a diverse and inclusive community of learners. By embracing diversity in all its forms, including cultural, ethnic, and socioeconomic diversity, the program enriches the learning experience and prepares students to thrive in an increasingly globalized and interconnected world.

Ultimately, the MCA programme is more than just a degree it is a transformative educational experience that prepares students to become leaders, innovators, and change makers in the field of computer science and beyond. Whether students aspire to pursue careers in software development, data analytics, cyber security, academia, or entrepreneurship, the MCA programme equips them with the skills, knowledge, and mindset needed to succeed in today's digital economy and shape the future of technology.

A. Programme’s Mission and Objectives

Mission:

The MCA Programme is dedicated to:

- Delivering top-notch education blending theory and practical skills in computer science.
- Equipping students with critical problem-solving abilities and technical expertise.
- Providing hands-on experiences through internships and projects.
- Cultivating innovation and entrepreneurial spirit among students.
- Fostering a diverse and inclusive learning environment.

- Preparing students for global challenges and opportunities.
- Instilling ethical values and social responsibility.
- Promoting lifelong learning and adaptability to technological changes.
- Encouraging community engagement and societal contribution.
- Offering ongoing support and resources to alumni for professional growth.

Objectives:

The objectives of the MCA programme are multifaceted, aiming to provide a comprehensive education in computer science while preparing students for successful careers in the IT industry. Key goals include delivering a high-quality curriculum that meets global standards, equipping students with technical proficiency and practical skills through hands-on experiences, and offering specialized knowledge in areas such as software engineering and artificial intelligence. Additionally, the program focuses on fostering critical thinking, communication, and collaboration skills, promoting ethical conduct and professional integrity, and instilling a commitment to lifelong learning. With an emphasis on career readiness and a global perspective, the program seeks to prepare students to thrive in diverse professional environments and contribute meaningfully to the advancement of technology and innovation. Ongoing feedback and continuous improvement processes ensure that the program remains responsive to the evolving needs of students, industry, and society.

B. Relevance of the Programme with HEI's Vision and Goals

The vision and mission of HEI, MATA TRIPURA SUNDARI OPEN UNIVERSITY, of Agartala are:

Vision:

The vision for the MCA programme in our institution is to provide a transformative educational experience that empowers students to excel in the dynamic field of computer science and information technology. Through a rigorous curriculum blending theory and practical application, the program aims to equip students with the skills, knowledge, and mindset needed to thrive in today's digital landscape. Emphasizing innovation, collaboration, and diversity, the program fosters a culture of excellence and prepares students to become leaders, innovators, and change makers in the IT industry and beyond. By embracing lifelong learning and continuous improvement, the program remains responsive to the evolving needs of students, industry, and society, ensuring that graduates are well-prepared to make a positive impact in the world through technology.

Goals:

The relevance of the MCA programme to Higher Education Institutions' (HEIs) goals lies in its alignment with the broader mission and objectives of the institution. Here's how the program can contribute to the HEI's goals:

- **Academic Excellence:** The MCA programme can enhance the reputation of the institution by delivering high-quality education in computer science and information technology, attracting talented students and faculty, and producing graduates who excel in their careers.

- **Research and Innovation:** The program can contribute to the institution's research goals by fostering a culture of innovation and inquiry among students and faculty, engaging in cutting-edge research projects, and collaborating with industry partners to develop innovative solutions to real-world problems.
- **Community Engagement:** Through service-learning initiatives, collaborative projects, and outreach programs, the MCA programme can strengthen the institution's ties to the community, demonstrating its commitment to social responsibility and civic engagement.
- **Global Perspective:** By exposing students to global perspectives, multicultural environments, and international collaborations, the program can help the institution fulfill its goal of preparing students to thrive in an increasingly interconnected world.
- **Employability and Career Development:** The program can support the institution's goal of enhancing graduates' employability and career prospects by providing them with the skills, knowledge, and experience needed to succeed in the workforce and adapt to changing industry demands.
- **Diversity and Inclusion:** By promoting diversity, equity, and inclusion within the program, the institution can create a more welcoming and inclusive campus environment, reflecting its commitment to social justice and equality.
- **Continuous Improvement:** Through ongoing assessment, evaluation, and feedback mechanisms, the program can contribute to the institution's goal of continuous improvement, ensuring that it remains responsive to the evolving needs of students, industry, and society.

Overall, the MCA programme can serve as a key driver of the institution's goals, enhancing its reputation, impact, and contribution to the broader community.

C. Nature of Prospective Target Group of Learners

The Distance Education of MATA TRIPURA SUNDARI OPEN UNIVERSITY (MTSOU) shall target the working professionals, executives as well as those who cannot attend a full-time programme due to prior occupation or other assignments. The candidates desirous of taking admission in Master in Computer Application (M.CA) programme shall have to meet the eligibility norms as follows –

- To obtain admission in M.CA programme offered through ODL mode, the learner must have completed graduation in science stream or equivalent course.
- The ODL MCA programme offered by MATA TRIPURA SUNDARI OPEN UNIVERSITY (MTSOU) caters the needs of diverse groups of undergraduate learners from all disciplines located in diverse regions and social structures such as learners from a low level of disposable income, rural dwellers, women and minorities who have little access to formal institutions of higher learning.

D. Appropriateness of Programme to be conducted in ODL mode to acquire specific skills and competence

The University has identified the following **Programme Outcomes (PO)** and **Programme Specific Outcomes (PSO)** as acquisition of specific skills and competence in MCA Programme.

Programme Outcomes (PO's)

After completing the MCA programme, students will be able to:

PO1 (Critical Thinking): Develop proficiency of techniques, knowledge of vocabulary, to show creative, critical and philosophical thinking of work.
PO2 (Professional Skill): Develop understanding of professional needs, responsibilities, and requirements as an art professional.
PO3 (Decision Making): Show potential in providing creative solutions to communication of complex phenomena of print media such as books, magazines and newspaper, able to depict and perform in group settings.
PO4 (Ethics): Recognize different value systems including their own, understand the moral dimensions of their decisions, and accept responsibility for them.
PO5 (Environment and Sustainability): Understand the issues of environmental contexts and sustainable development.
PO6 (Self-directed and Life-long Learning): Acquire the ability to engage in independent and life-long learning.

Programme Specific Outcomes (PSOs)

PSO1: Enrich the knowledge in the areas like Artificial Intelligence, Web Services, Cloud Computing, Paradigm of Programming language, Design and Analysis of Algorithms, Database Technologies Advanced Operating System, Mobile Technologies, Software Project Management and core computing subjects. Choose to study any one subject among recent trends in IT provided in the optional subjects.
PSO2: Students understand all dimensions of the concepts of software application and projects.
PSO3: Students understand the computer subjects with demonstration of all programming and theoretical concepts with the use of ICT.
PSO4: Developed in-house applications in terms of projects.

E. Instructional Design

The MCA programme is structured into four semesters, with a minimum credit requirement of 80 to obtain the degree. In ODL (Open and Distance Learning) mode from MATA TRIPURA SUNDARI OPEN UNIVERSITY, the minimum time period for completing the MCA degree is two years, while the maximum allowable time period is four years.

MATA TRIPURA SUNDARI OPEN UNIVERSITY
Master of Computer Application (MCA)
Course structure and Syllabi
(2025-26)

Total Program Credits for MCA

S.No.	Semester	Credits
1	I	20
2	II	20
3	III	20
4	IV	20
	Total	80

Induction Program

Conduction of induction program as per UGC/Regulatory Body's guideline.

Evaluation Scheme

SEMESTER – I						
S.No.	Course Code	Course Name	Credit	Continuous Assessment Marks	Term End Exam	Grand Total
				Max. Marks	Max. Marks	
1	CSM-6111	Data Communication & Computer Networks	4	30	70	100
2	CSM-6112	Computer Organization & Architecture	3	30	70	100
3	ENM-6101	Professional Communication	2	30	70	100
4	CSM-6113	Discrete Mathematics	4	30	70	100
5	CSM-6114	Accountancy and Financial Management	3	30	70	100
6	CSM-6115	Programming with 'C'	2	30	70	100
7	CSM-6151	Programming with 'C' Lab	2	30	70	100
TOTAL			**20**	**210**	**490**	**700**

SEMESTER – II						
S.No.	**Course Code**	**Course Name**	**Credit**	**Continuous Assessment Marks**	**Term End Exam**	**Grand Total**
				Max. Marks	**Max. Marks**	
1	CSM-6211	Web Programming	4	30	70	100
2	CSM-6212	Advance Cyber Security	3	30	70	100
3	CSM-6213	Management Information & system	3	30	70	100
4	CSM-6214	Design & Analysis of Algorithm	3	30	70	100
5	CSM-6215	Data Structure using C++	3	30	70	100
6	CSM-6251	DAA and Web Programming Lab	2	30	70	100
7	CSM-6252	Data Structure using C++ Lab	2	30	70	100
TOTAL			**20**	**210**	**490**	**700**

Note: The students taking an exit from programme after securing 40 credit of the first and second semester shall be awarded a PG Diploma in Computer Applications.

SEMESTER – III						
S.No.	**Course Code**	**Course Name**	**Credit**	**Continuous Assessment Marks**	**Term End Exam**	**Grand Total**
				Max. Marks	**Max. Marks**	
1	CSM-7111	Artificial Intelligence and Machine Learning	3	30	70	100
2	CSM-7112	Data Science using R Programming	3	30	70	100
3	CSM-7113	OOP's Technologies and Java Programming	3	30	70	100
4	CSM-7114	Advanced DBMS	3	30	70	100
5		Elective – I	4	30	70	100
6	CSM-7151	Java Programming Lab	2	30	70	100
7	CSM-7152	Data Science using R Programming Lab	2	30	70	100
TOTAL			**20**	**210**	**490**	**700**

Elective – I

Any One Course from the Following 03 (Electives Course – CSM – 7115 / CSM – 7116 / CSM – 7117)						
1	CSM-7115	Soft Computing Techniques	4	30	70	100
2	CSM-7116	Data Warehousing and Data Mining	4	30	70	100
3	CSM-7117	Cloud Computing and Internet of Things	4	30	70	100

SEMESTER – IV						
S.No.	**Course Code**	**Course Name**	**Credit**	**Continuous Assessment Marks**	**Term End Exam**	**Grand Total**
				Max. Marks	**Max. Marks**	
1	CSM-7211	Big Data Analytics	3	30	70	100
2	CSM-7212 / CSM-7213	Elective - II	3	30	70	100
3	CSM-7214	Natural Language Processing	3	30	70	100
4	CSM-7215	Python Programming	2	30	70	100
5	CSM-7216	Quantum Computing	3	30	70	100
6	CSM-7251	Python Programming Lab	2	30	70	100
7	CSM-7291	**Project**	4	30	70	100
TOTAL			**20**	**210**	**490**	**700**

Elective – II

Any One Course from the Following 02 (Electives Course – CSM – 7212 / CSM – 7213)						
1.	CSM-7212	Mobile Computing	3	30	70	100
2.	CSM-7213	Deep Learning	3	30	70	100

Note: The students who want to undertake two year PG Programme will be awarded Master in Computer Applications (MCA) degree upon securing 80 credits.

SEMESTER – I

Course Code: CSM – 6111 **Credit: 4**
Course Name: Data Communication & Computer Networks

Course objectives:

- To provide students with a comprehensive understanding of the fundamental principles of data communication and computer networks, including the various types of networks, network topologies, protocols, and standards.
- To equip students with the knowledge and skills necessary to design, implement, and manage computer networks, covering both local area networks (LANs) and wide area networks (WANs), as well as emerging network technologies.
- To develop students' abilities to analyze and evaluate network performance, identify potential security threats, and implement effective measures to enhance network security and reliability.
- To provide practical experience through hands-on labs and projects, enabling students to use network simulation tools, configure network devices, troubleshoot network issues, and apply theoretical knowledge in real-world scenarios.
- To introduce students to advanced topics in data communication and networking, such as wireless networks, cloud computing, Internet of Things (IoT), and next-generation networking technologies, preparing them for future trends and innovations in the field.

Course Outcomes (CO's):
On successful completion of this practical lab session, students will be able to:
1. Understand the fundamental principles and concepts of data communication and computer networks, including network architectures, protocols, and transmission technologies.
2. Develop proficiency in designing, implementing, and troubleshooting computer networks to meet specific communication requirements.
3. Gain knowledge of network security principles and techniques, enabling the identification and mitigation of potential security threats and vulnerabilities.
4. Acquire skills in analyzing network performance and optimizing network resources for efficient data transmission.
5. Prepare students for professional roles in network engineering, system administration, and cyber security through practical experience and theoretical understanding of data communication and computer networks.

Block-I: Fundamental Concepts & Network Architecture
Unit-1: Introduction to Computer Networks – Goals, Applications, and Benefits
Unit-2: Network Structures, Architectures, and Topologies
Unit-3: OSI and TCP/IP Reference Models – Layers, Functions, and Protocols
Unit-4: Physical Layer – Transmission Media, Switching Techniques, ISDN, and Terminal Handling

Block-II: Data Link Layer & Network Layer
Unit-5: Channel Allocation – LAN Protocols, ALOHA Variants (Pure & Slotted ALOHA)

Unit-6: Medium Access Control – CSMA/CD, CSMA/CA, Collision-Free Protocols, IEEE Standards (Ethernet, Wi-Fi, FDDI)
Unit-7: Data Link Layer – Framing, Error Detection & Correction, Sliding Window Protocols, HDLC
Unit-8: Network Layer Functions – Addressing, Routing Algorithms, Congestion Control

Block-III: Transport Layer & Network Services
Unit-9: IP Addressing & Networking – IPv4, IPv6, Subnetting, CIDR, Internetworking
Unit-10: Transport Layer – Functions, Protocols, and Design Issues
Unit-11: Connection-Oriented & Connectionless Services, TCP Window Management
Unit-12: User Datagram Protocol (UDP) and Transmission Control Protocol (TCP)

Block-IV: Application Layer & Network Security
Unit-13: Flow Control, Error Control, and Quality of Service (QoS) in Transport Layer
Unit-14: Network Security Fundamentals – Encryption, Hashing, Firewalls, Intrusion Detection
Unit-15: Secure Communication – DES, AES, RSA, Digital Signatures, Domain Name System (DNS)
Unit-16: Internet Protocols & Services – SMTP, FTP, HTTP, SNMP, Cryptography & Compression Techniques

Books Recommended/Suggested Reading:

1. A. S Tanenbaum, “Computer Networks, 3rd Edition”, PHI
2. W. Stallings, “Data and Computer Communication”, Macmillan Press
3. Comer, “Computer Networks & Internet”, PHI.
4. Comer, “Internetworking with TCP/IP”, PHI

Course Code: CSM – 6114 **Credit: 3**

Course Name: Accountancy and Financial Management

Course Objective:

- To provide students with a solid understanding of fundamental financial principles and concepts, including financial statements, accounting principles, and financial analysis techniques.
- To teach students how to prepare and interpret financial statements, including balance sheets, income statements, and cash flow statements, and analyse financial performance using ratios and other tools.
- To introduce students to managerial accounting concepts and techniques, including cost analysis, budgeting, variance analysis, and decision-making using relevant financial information.
- To explore topics related to corporate finance, such as capital budgeting, risk and return, cost of capital, and capital structure decisions, enabling students to evaluate investment opportunities and make informed financial decisions.
- To familiarize students with financial management practices and strategies, including financial planning, working capital management, risk management, and corporate governance, to ensure efficient use of financial resources and maximize shareholder value.

Course Outcomes:

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

1. Familiarize the principles and concepts accounting which involved in business transactions
2. Enable to prepare trial balance, bank reconciliation statement, identify and rectify the errors in entries.
3. Enable to preparing final accounts and financial statement.
4. Have an knowledge about accounting standards to prepare effective and ethical financial statement.

Block-I: Overview of Accounting

Unit-1: Accounting concepts, conventions and principles;
Unit-2: Accounting Equation, International Accounting principles and standards;
Unit-3: Matching of Indian Accounting Standards with International Accounting Standards.

Block-II: Mechanics of Accounting

Unit-4: Double entry system of accounting, journalizing of transactions;
Unit-5: Preparation of final accounts, Profit & Loss Account, Profit & Loss Appropriation account and Balance Sheet,
Unit-6: Policies related with depreciation, inventory and intangible assets like copyright, trademark, patents and goodwill.

Block-III: Analysis of financial statement

Unit-7: Ratio Analysis- solvency ratios, profitability ratios, activity ratios, liquidity ratios, market capitalization ratios
Unit-8: Common Size Statement
Unit-9: Comparative Balance Sheet and Trend Analysis of manufacturing, service & banking organizations.

Block-IV: Funds Flow Statement & Cash Flow Statement

Unit-10: Meaning, Concept of Gross and Net Working Capital, Preparation of Schedule of Changes in Working Capital.
Unit-11: Preparation of Funds Flow Statement and its analysis;
Unit-12: Various cash and non-cash transactions, flow of cash, preparation of Cash Flow Statement and its analysis.

Books Recommended/Suggested Reading:

1) Narayanswami - Financial Accounting: A Managerial Perspective (PHI, 2 Edition)
2) Mukherjee - Financial Accounting for Management (TMH, Edition)
3) Ramchandran & Kakani - Financial Accounting for Management (TMH, 2 Edition)
4) Ghosh T P - Accounting and Finance for Managers (Taxman, 1 Edition)
5) Maheshwari S.N & Maheshwari S K – An Introduction to Accountancy (Vikas, 9 Edition)
6) Ashish K. Bhattacharya- Essentials of Financial Accounting (PHI, New Delhi)
7) Ghosh T.P- Financial Accounting for Managers (Taxman, 3 Edition)
8) Maheshwari S.N & Maheshwari S K – A text book of Accounting for Management (Vikas, 1 Edition)
9) Gupta Ambrish - Financial Accounting for Management (Pearson Education, 2 Edition)
10) Chowdhary Anil - Fundamentals of Accounting and Financial Analysis (Pearson Education, 1 Edition).

Course Code: ENM – 6101 **Credit: 2**
Course Name: Professional Communication

Course Objective:

- To develop students' ability to write clear, concise, and professional documents such as emails, reports, proposals, and other business correspondence, ensuring they can convey information effectively in written form.
- To enhance students' oral communication skills, enabling them to deliver engaging and informative presentations, participate in meetings, and communicate confidently and persuasively in professional settings.
- To foster strong interpersonal communication skills, including active listening, empathy, conflict resolution, and teamwork, ensuring students can build and maintain productive working relationships.
- To equip students with the skills to use various digital communication tools and platforms effectively, understanding the nuances of virtual communication and the importance of digital etiquette.
- To prepare students for the global business environment by developing their ability to communicate effectively across cultures, understanding cultural differences, and adapting their communication style to diverse audiences.

Course Outcomes:

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

1. Demonstrate an understanding of the principles and theories of professional communication.
2. Write clear, concise, and persuasive business documents, such as e-mails, job application, resume and cover letters)
3. Deliver effective presentations using appropriate visual aids and speaking techniques.
4. Collaborate with others in a professional setting, demonstrating effective teamwork and leadership skills.

Block 1: **Foundations of Professional Communication**

Unit 1: Introduction to Professional Communication
Unit 2: Written Communication Skills (E-mail, Job application, Resume and Cover letter)
Unit 3: Verbal Communication Skills
Unit 4: Interpersonal and Team Communication
Unit 5: Digital Communication Tools and Impact of technology

Block 2: Advanced Professional Communication Strategies

Unit 6: Visual and Multimedia Communication
Unit 7: Effective Presentation strategies: Introduction, Defining purpose, Analyzing Audience and Locale
Unit 8: Understanding Nuances of Delivery
Unit 9: Group discussion as a part of selection process
Unit 10: Ethics and Legal Considerations in Professional Communication

Books Recommended/Suggested Reading:

1. "Improve Your Writing ed. V.N. Arora and Laxmi Chandra, Oxford Univ. Press, New Delhi.
2. Technical Communication – Principles and Practices by Meenakshi Raman &Sangeeta Sharma, Oxford Univ. Press 2007, New Delhi.
3. Business Correspondence and Report Writing by Prof. R.C. Sharma & Krishna Mohan, Tata McGraw Hill & Co. Ltd., New Delhi.
4. How to Build Better Vocabulary by M. Rosen Blum, Bloomsbury Pub. London.
5. Word Power Made Easy by Norman Lewis, W. R. Goyal Pub. & Distributors; Delhi.
6. Developing Communication Skills by Krishna Mohan, Meera Banerji- Macmillan India Ltd. Delhi.
7. Manual of Practical Communication by L.U.B. Pandey & R.P. Singh; A.I.T.B.S.
8. Publications India Ltd.; Krishan Nagar, Delhi.

Course Code: CSM – 6113 **Credit: 4**
Course Name: Discrete Mathematics

Course Objective:

- To provide students with a solid foundation in the fundamental concepts of discrete mathematics, including sets, relations, functions, logic, and proof techniques.
- To teach students the principles of combinatorics and graph theory, enabling them to solve problems related to counting, graph traversal, graph coloring, and network flows.
- To introduce students to basic algorithms and their complexity, helping them understand how to analyze the efficiency of algorithms and solve problems using algorithmic thinking.
- To develop students' knowledge of discrete structures such as lattices, Boolean algebras, and algebraic structures, and their applications in computer science and related fields.
- To enhance students' mathematical reasoning and problem-solving skills, equipping them with the ability to apply discrete mathematics concepts to real-world problems and theoretical computer science.

Course Outcomes:

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

1. Verify the correctness of an argument using symbolic logic and truth tables.
2. Construct proofs using mathematical induction.
3. Define algebraic structures and other group related terms
4. Analyse the concept lattice and related properties
5. Introduce the language and related terms

Block I: Relation

Unit 1: Type and compositions of relations, Equivalence relations, Partial order relation.
Unit 2: Function: Types, Composition of function, recursively defined function
Unit 3: Mathematical Induction: Piano's axioms, Discrete Numeric Functions and Generating functions, Asymptotic Behavior of numeric functions
Unit 4: Algebraic Structures, Semi group, Monoid, Group, Abelian group, properties of group, Subgroup, Cyclic group, Cosets, Permutation groups, Homomorphism, Isomorphism and Automorphism of groups.

Block II: Propositional Logic

Unit 5: Preposition, First order logic, Basic logical operations, Tautologies, Contradictions, Algebra of Proposition, Logical implication, Logical equivalence, Predicates and quantifiers.
Unit 6: Lattices: Introduction, ordered set, Hasse diagram of partially ordered set.
Unit 7: Consistent enumeration, Isomorphic ordered set, well ordered set, Lattices.
Unit 8: Properties of lattices, Bounded lattices, Distributive lattices and Complemented lattices.

Block III: Introduction to defining language

Unit 9: Introduction to language: Kleene Closure, Arithmetic expressions
Unit 10: Chomsky Hierarchy, Regular expressions, Generalized Transition graph.

Block IV: Regular expression

Unit 11: Deterministic finite automata, strings & transition function

Unit 12: Non-deterministic finite automata, with epsilon transitions & languages

Block V: Non-regular language

Unit 13: Push Down Automata & CFG to PDA Conversion

Unit 14: Turing machine & chomsky hierarchy

Books Recommended/Suggested Reading:

1. J. P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw-Hill,1997.

2. S. Lipschutz and M. L. Lipson, Schaum's Outline of Theory and Problems of Discrete Mathematics, 2nd Ed., Tata McGraw-Hill,1999.

3. K. H. Rosen, Discrete Mathematics and its applications, Tata McGraw-Hill, 6th Ed., 2007.

4. David Liben-Nowell, Discrete Mathematics for Computer Science, Wiley publication, July 2017.

5. Hari Kishan, Shiv Raj Pundir, Discrete Mathematics, Pragati Prakashan, Meerut.

Course Code: CSM – 6114 **Credit: 3**

Course Name: Accountancy and Financial Management

Course Objective:

- To provide students with a solid understanding of fundamental financial principles and concepts, including financial statements, accounting principles, and financial analysis techniques.
- To teach students how to prepare and interpret financial statements, including balance sheets, income statements, and cash flow statements, and analyse financial performance using ratios and other tools.
- To introduce students to managerial accounting concepts and techniques, including cost analysis, budgeting, variance analysis, and decision-making using relevant financial information.
- To explore topics related to corporate finance, such as capital budgeting, risk and return, cost of capital, and capital structure decisions, enabling students to evaluate investment opportunities and make informed financial decisions.
- To familiarize students with financial management practices and strategies, including financial planning, working capital management, risk management, and corporate governance, to ensure efficient use of financial resources and maximize shareholder value.

Course Outcomes:

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

1. Familiarize the principles and concepts accounting which involved in business transactions
2. Enable to prepare trial balance, bank reconciliation statement, identify and rectify the errors in entries.
3. Enable to preparing final accounts and financial statement.
4. Have an knowledge about accounting standards to prepare effective and ethical financial statement.

Block-I: Overview of Accounting

Unit-1: Accounting concepts, conventions and principles;
Unit-2: Accounting Equation, International Accounting principles and standards;
Unit-3: Matching of Indian Accounting Standards with International Accounting Standards.
Unit-4: Double entry system of accounting, journalizing of transactions;

Block-II: Mechanics of Accounting

Unit-5: Preparation of final accounts, Profit & Loss Account, Profit & Loss Appropriation account and Balance Sheet,
Unit-6: Policies related with depreciation, inventory and intangible assets like copyright, trademark, patents and goodwill.
Unit-7: Ratio Analysis- solvency ratios, profitability ratios, activity ratios, liquidity ratios, market capitalization ratios
Unit-8: Common Size Statement

Block-III: Analysis of financial statement

Unit-9: Comparative Balance Sheet and Trend Analysis of manufacturing, service & banking organizations.

Block-IV: Funds Flow Statement

Unit-10: Introduction to Funds Flow Statement: Overview of Funds Flow Statement: Understanding the purpose and significance of funds flow statement in financial analysis. Conceptual Framework: Explaining the conceptual framework of funds flow statement and its relationship with other financial statements.

Unit-11: Investing Activities: Cash Flow from Investments: Evaluating cash flows related to investing activities such as purchase and sale of property, plant, and equipment, and investments in securities. Capital Expenditure Analysis, Return on Investment (ROI.
Unit-12: Meaning, Concept of Gross and Net Working Capital, Preparation of Schedule of Changes in Working Capital.
Unit-13: Preparation of Funds Flow Statement and its analysis;

Block-V: Cash Flow Statement
Unit-14: Introduction to Cash Flow Statement: Overview of Cash Flow Statement: Understanding the purpose, importance, and objectives of the cash flow statement in financial reporting. Conceptual Framework: Explaining the conceptual framework of the cash flow statement and its relationship with other financial statements, Regulatory Requirements.
Unit-15: Cash Flow from Operating Activities: Operating Cash Inflows: Identifying cash inflows from operating activities such as sales revenue, interest income, and dividends received. Operating Cash Outflows: Analyzing cash outflows from operating activities including payments to suppliers, employees, and creditors.
Unit-16: Cash Flow Ratios and Analysis, Liquidity Ratios: Calculating liquidity ratios such as the cash flow ratio, operating cash flow ratio, and cash flow coverage ratio to assess liquidity position. Solvency Ratios, Cash Flow Forecasting: Forecasting future cash flows based on historical data and industry.
Unit-17: Cash Flow Management Strategies, Working Capital Management, Implementing working capital management strategies to optimize cash flow, minimize liquidity risks, and enhance profitability. Investment Strategies: Developing investment strategies, Financing Strategies.
Unit-18: Cash Regulatory Compliance and Disclosure, Accounting Standards: Ensuring compliance with relevant accounting standards and regulatory requirements, Financial Reporting Practices.

Books Recommended/Suggested Reading:

1) Narayanswami - Financial Accounting: A Managerial Perspective (PHI, 2 Edition)
2) Mukherjee - Financial Accounting for Management (TMH, Edition)
3) Ramchandran & Kakani - Financial Accounting for Management (TMH, 2 Edition)
4) Ghosh T P - Accounting and Finance for Managers (Taxman, 1 Edition)
5) Maheshwari S.N & Maheshwari S K – An Introduction to Accountancy (Vikas, 9 Edition)
6) Ashish K. Bhattacharya- Essentials of Financial Accounting (PHI, New Delhi)
7) Ghosh T.P- Financial Accounting for Managers (Taxman, 3 Edition)
8) Maheshwari S.N & Maheshwari S K – A text book of Accounting for Management (Vikas, 1 Edition)
9) Gupta Ambrish - Financial Accounting for Management (Pearson Education, 2 Edition)
10) Chowdhary Anil - Fundamentals of Accounting and Financial Analysis (Pearson Education, 1 Edition).

Course Code: CSM – 6115 **Credit: 2**
Course Name: Programming with 'C'

Course Objectives:

- To provide students with a foundational understanding of programming concepts and logic, using the C programming language as a vehicle for learning basic programming constructs.
- To familiarize students with the syntax and semantics of the C programming language, including variables, data types, operators, control structures, functions, and arrays.
- To develop students' problem-solving skills by providing them with opportunities to apply programming concepts to solve a variety of real-world problems, through hands-on coding exercises and projects.
- To teach students how to debug and troubleshoot C programs effectively, including techniques for identifying and fixing common errors and logical flaws in code.
- To provide students with practical experience in writing, compiling, executing, and debugging C programs in a laboratory environment, reinforcing theoretical concepts through hands-on experimentation and coding exercises.

Course Outcome (CO's):

On successful completion of this course, students should be able to:

1. Demonstrate a thorough understanding of the syntax and semantics of the C programming language.
2. Apply fundamental programming constructs proficiently to solve a variety of computational problems.
3. Implement and manipulate data structures such as arrays, pointers, and linked lists effectively.
4. Perform file processing operations, including reading from and writing to files, with accuracy and efficiency.
5. Develop well-structured and efficient C programs, showcasing strong problem-solving and programming skills.

Block I: Algorithmic Process & Basic of 'C' Programming

Unit–1: Algorithms, General Approaches & Analysis, Program and Programming Language, Fundamental Stages of Problem Solving, Feature of Programming Language, Flow Charts.

Unit–2: Learning outcomes, Program and Programming Language, Introduction to C Language, Programming Format of C.

Unit–3: Creating a C Program, Compilation process in C Program, Link and Running C Program, Diagrammatic Illustration.

Block II: Operator and Expressions of 'C', Control Flow Mechanisms

Unit–4: Building Blocks – Character set of C, C Tokens, Keywords and Identifiers of the C. Fundamental elements of 'C' – Data Types in C, Variables.

Unit–5: Logical and Relational – Operators in 'C', Expressions in 'C' and Types Conversions in Expressions.

Unit–6: Key Terminologies, Design Control Statements, Loop Control Statements and Exit Function.

Unit–7: Declaring & Accessing Data Elements, Arrays Declaration, Initialization and Passing Functions.

Block III: Strings, Tools for Modular Programming and Pointers

Unit–8: Essential Techniques & Functions, Declaration and
Initialization of Strings, Overview and Applications.
Unit–9: Functions Prototypes, Calling a Function, Return Statement, Sets of Variables & Storage Classes and Recursion.
Unit–10**:** Handle Variables and Parameters, Pointer and their Characteristics, Passing Pointers to Functions and Pointers and Strings.

Block IV: Multiple Data Elements, Pre-processors Directives and Files
Unit–11: Declaration of Structures, Accessing the Members of a Structure, Initializing, Function Arguments and Pointers to Structures.
Unit–12: Defining of Unions, Initialization of Unions and Accessing the Members of an Union.
Unit–13: Translation Phase, 'C' Pre-processor, Implement Constants, Reading from other files and Conditional Selection of code and Pre-Processor Commands.
Unit–14: File Handling in C using file Pointers, Input and Output using file Pointers, Sequential Vs Random Access Files and Unbuffered I/O – The UNIX File Routines.

Books Recommended/Suggested Reading:

1. Kamthane A. N. and Kamthane A. A.; Programming in C, Pearson Education India.
2. Reema Thareja; Computer Fundamentals and Programming in C, Oxford University Press.
3. Dey P. and Ghosh M.; Programming in C, Oxford University Press.
4. Kernighan B. W. and Dennis M. R.; The C Programming Language, Pearson Education India.
5. Kanetkar Y. P.; Let us C, BPB Publications.
6. Jones J. A. and Harrow K.; Problem solving with C, Pearson Education India.

Course Code: CSM – 6151 **Credit: 2**
Course Name: Programming with 'C' Lab

Course Objectives:

- To develop problem-solving skills using C programming.
- To understand fundamental concepts of variables, data types, and operators in C.
- To implement control structures, loops, and functions for efficient coding.
- To apply arrays, pointers, and structures for data manipulation.
- To enhance debugging and error-handling techniques in C programs.

Course Outcome (CO's):

On successful completion of this course, students should be able to:

1. Demonstrate proficiency in writing, compiling, and executing C programs.
2. Apply control structures, loops, and functions to solve computational problems.
3. Utilize arrays, pointers, and structures for effective memory management.
4. Develop modular programs using functions and file handling techniques.
5. Debug and optimize C programs for efficiency and correctness.

Programming Lab

- Introduction (Overview of the Lab)
- Objectives
- Overall Directions
- Structure of 'C' Program
- Salient Features of C
- 'C' Program development Environment
    - Phase-I: Creating a Program
    - Phase-II&III: Preprocessing and Compiling a 'C' Program
    - Install Visual Studio Code on Windows
- How to design/develop Program
- Structure of 'C' Program
- Compile and Run 'C' Program
- Practice Sessions (Session 1 to Session 5)

Programs:

1. Write a C program to find roots of a quadratic equation.
2. Write a C program to find the total no. of digits and the sum of individual digits of a positive integer.
3. Write a C program to generate the Fibonacci sequence of first N numbers.
4. Write a C program to compute area of a circle, Square and Rectangle when all the dimension are given.
5. Write a C program to input two matrices and perform matrix multiplication on them.
6. Write a C program to check whether the given string is palindrome or not without using Library functions.
7. Write a C program to count the number of lines and words in a given file.

8. Write a C program to generate prime numbers in a given range using user defined function.
9. Write a C program to find factorial of a given number using recursive function.
10. Write a C program to maintain a record of n student details using an array of structures with four fields - Roll number, Name, Marks and Grade. Calculate the Grade according to the following conditions.

    Marks Grade

    >= 80 A

    >= 60 B

    >= 50 C

    >= 40 D

    < 40 E

    Print the details of the student, given the student roll number as input.

Books Recommended/Suggested Reading:

7. Kamthane A. N. and Kamthane A. A.; Programming in C, Pearson Education India.
8. Reema Thareja; Computer Fundamentals and Programming in C, Oxford University Press.
9. Dey P. and Ghosh M.; Programming in C, Oxford University Press.
10. Kernighan B. W. and Dennis M. R.; The C Programming Language, Pearson Education India.
11. Kanetkar Y. P.; Let us C, BPB Publications.
12. Jones J. A. and Harrow K.; Problem solving with C, Pearson Education India.

SEMESTER – II

Course Code: CSM - 6211 **Credit: 4**

Course Name: Web Programming

Course Objective:

- To introduce students to fundamental web technologies, including HTML, CSS, and JavaScript, and their role in creating dynamic and interactive web pages.
- To teach students how to use JavaScript to add interactivity and functionality to web pages, including form validation, DOM manipulation, and event handling.
- To familiarize students with server-side scripting languages such as PHP, or Node.js, and their use in processing user requests, accessing databases, and generating dynamic content.
- To enable students to integrate databases into web applications, including connecting to databases, executing queries, and displaying query results dynamically on web pages.
- To introduce students to popular web development frameworks such as React, Angular, or Vue.js, and teach them how to use these frameworks to streamline the development process and build scalable web applications.

Course Outcomes:

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

1. Students will gain proficiency in HTML, CSS, and JavaScript, as well as other web technologies and frameworks commonly used in web design, enabling them to create responsive and interactive websites.
2. Students will develop an understanding of design principles such as layout, typography, color theory, and usability, and learn to apply them effectively in designing visually appealing and user-friendly websites.
3. Students will learn techniques for creating responsive web designs that adapt to different screen sizes and devices, ensuring optimal user experience across desktops, tablets, and smartphones.
4. Students will learn to incorporate interactive elements such as animations, transitions, and dynamic content using JavaScript and other scripting languages, enhancing user engagement and interactivity.
5. Students will learn techniques for ensuring cross-browser compatibility and testing websites across different web browsers, platforms, and devices, to ensure consistent performance and functionality.
6. Through hands-on projects and exercises, students will apply their knowledge of web design principles and technologies to create real-world websites, gaining practical experience and developing their design and development skills.

Block-I: Introduction

Unit-1: What is Markup Language Basic Structure of HTML

Unit-2 : Difference Between HTML and XHTML

Unit-3: Head Section and Elements of Head Section Meta Tags

Unit-4: CSS Tags Script Tag Table Tag Div Tag Header Tags Paragraph, pan, Pre Tags

Block-II: Designing Pages with HTML & CSS

Unit-5: Anchor Links and Named Anchors Image Tag Object Tag frame Tag Forms Form Tag attributes of Form POST and GET Method

Unit-6: Field set and Legend Text input, Text area Checkbox and Radio Button Dropdown.

Unit-7: Dynamic HTML, Document Object Model, Features of DHTML,

Unit-8: CSSP (Cascading Style Sheet Positioning)

Unit-9: JSSS (Java Script assisted Style Sheet

Block-III: Java Script

Unit-10: Objects,

Unit-11: Methods, Events and Functions,

Unit-12: Tags, Operators, Data Types,

Unit-13: Literals and Type Casting in JavaScript,

Unit-14: Programming Construct Array and Dialog Boxes

Block-IV: Front Page

Unit-15: Front Page Basics,

Unit-16: Web Terminologies,

Unit-17: Phases of Planning and Building Web Sites,

Unit-18: the FTP, HTTP and WPP.

Books Recommended/Suggested Reading:

1. Teach Yourself HTML 4.0 with XML, DHTML and Java Script by Stephanie, Cottrell, Bryant; IDG Books India Pvt. Ltd., New Delhi
2. Dynamic Web Publishing – Unleashed Tech Media

Course Code: CSM - 6212 **Credit: 3**

Course Name: Advance Cyber Security

Course objectives:

- To provide students with advanced knowledge and skills in analysing cybersecurity threats, including malware analysis, threat intelligence gathering, and reverse engineering techniques.
- To equip students with the ability to respond effectively to cybersecurity incidents, including incident detection, containment, eradication, and recovery, as well as forensic investigation techniques for collecting and analysing digital evidence.
- To teach students how to design, implement, and manage security operations centres (SOCs), including security monitoring, log analysis, and incident management processes, using industry-standard tools and techniques.
- To enable students to identify, assess, and mitigate vulnerabilities in systems and networks, including vulnerability scanning, penetration testing, and patch management strategies.
- To familiarize students with cybersecurity governance frameworks, regulations, and compliance standards, including GDPR, HIPAA, ISO 27001, and NIST Cybersecurity Framework, and how to implement cybersecurity controls to achieve compliance.

Course Outcomes (CO's):

On successful completion of this practical lab session, students will be able to:

1. Apply IT ACT (Cyber law) to the given case/problem and infer from the given case and analyze the gap if it exists.
2. Analyze the working of cyber security principles in designing the system.
3. Develop a strategy (physical, logical or administrative controls) to mitigate the problem and articulate consequences on Society and National Economy.
4. Examine relevant network defense / web application tools to solve given cyber security problems and evaluate its suitability.
5. Investigate the influence of Block chain technology for the cyber security problem and evaluate its role.

Block I: Introduction to Cybercrime and Laws

Unit 1: Introduction, Cybercrime: Definition and Origins of the word, Cybercrime and information Security.

Unit 2: Cybercriminals, Classifications of Cyber Crimes.

Unit 3: Criminals Plan Them – Introduction, How Criminals Plan the Attacks.

Unit 4: Cybercafé and Cybercrimes, Botnets, Attack Vector.

Unit 5: Basic text markup; Images; Hypertext Links; Lists.

Block II: Tools and Methods used in Cybercrime

Unit 6: Introduction, Proxy Server and Anonymizers, Password Cracking. Key loggers and Spyware, Virus and Worms, Trojan and backdoors, Steganography.

Unit 7: DOS and DDOS attack.

Block III: Phishing and Identity Theft

Unit 8: Introduction, Phishing – Methods of Phishing.

Unit 9: Phishing Techniques, Phishing Toolkits and Spy Phishing. Identity Theft – PII.

Unit 10: Types of Identity Theft, Techniques of ID Theft.

Unit 11: Digital Forensics Science, Need for Computer Cyber forensics.

Block IV: Command lines and Backtracking

Unit 12: Advanced Unix Command Line Tools and Backtrack Evolution

- Essential Unix/Linux command line utilities for cybersecurity
- File manipulation, user management, networking commands
- Introduction to **Kali Linux** (successor to Backtrack)
- Common tools in Kali for penetration testing and ethical hacking

Unit 13: Cross-Platform Terminal Environments

- Use of **Mac Ports** for Unix-like command line utilities on macOS
- Introduction to **Cygwin** for creating a Linux-like environment on Windows
- Comparison of native and emulated command-line tools across platforms

Unit 14: Windows Command Line for Cybersecurity

- Introduction to **Windows PowerShell**: Scripting and automation for security tasks
- Core PowerShell commands for system administration and auditing
- Introduction to **Netcat (nc)**: Installation and basic commands

Unit 15: Network Tools and Secure Shell

- **Netcat** for port scanning, banner grabbing, and reverse shells
- Real-world use cases in penetration testing
- **SSH (Secure Shell)**: Secure remote login, key-based authentication
- Configuring and hardening SSH for secure communication

Block V: Network Defense tools and block chain technology

Unit 16: Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall.

Unit 17: Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls.

Unit 18: Network Address Translation (NAT) and Port Forwarding, the basic of Virtual Private Networks.

Books Recommended/Suggested Reading:

1. Anti-Hacker Tool Kit (Indian Edition) by Mike Shema, Publication McGraw Hill. (Chapters: 2, 7, 8, 11) 2014,4 th edition.

2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and Sunit Belpure, Publication Wiley 2011.
3. Yogesh Singh, “Software Testing”, Cambridge University Press, New York.
4. Marc Roper, “Software Testing”, McGraw-Hill Book Co., London.
5. Boris Beizer, “Software System Testing and Quality Assurance”, Van Nostrand Reinhold, New York.

Course Code: CSM - 6213 **Credit: 3**
Course Name: Management Information & System

Course objectives:

- To introduce students to the fundamentals of information systems, including their role in organizations, types of information systems, and their strategic importance for decision-making and competitive advantage.
- To provide students with knowledge and skills in database management systems (DBMS), including database design, implementation, querying, and administration, to support organizational data management needs.
- To familiarize students with enterprise resource planning systems and their role in integrating business processes across functional areas, including modules such as finance, human resources, supply chain management, and customer relationship management.
- To teach students how to use business intelligence tools and techniques to analyze organizational data, generate insights, and support decision-making processes, including data visualization, reporting, and predictive analytics.
- To enable students to understand the principles of information security, risk management, and cybersecurity, including threats, vulnerabilities, risk assessment, and mitigation strategies, to protect organizational assets and ensure compliance with regulatory requirements.

Course Outcomes (CO's):

On successful completion of this practical lab session, students will be able to:

1. Enhanced ability to leverage information technology for strategic decision-making and organizational performance optimization.
2. Proficiency in analyzing, designing, and managing information systems to meet organizational information needs and strategic objectives.
3. Mastery in utilizing information technology for strategic decision-making and organizational performance enhancement.
4. Proficiency in analyzing and designing information systems to meet organizational information requirements.
5. Ability to implement and manage information systems effectively to support business processes and operations.
6. Enhanced skills in leveraging data and business intelligence tools for informed decision-making and competitive advantage.

Block I: System Engineering, Information and Knowledge

Unit 1: System concepts, system control, types of systems, handling system complexity, Classes of systems, General model of MIS.

Unit 2: Need for system analysis, System analysis for existing system & new requirement, system development model, MIS & system analysis

Unit 3: Information concepts, classification of information, methods of data and information collection.

Unit 4: Value of information, information: A quality product, General model of a human as information processor, Knowledge.

Block II: Introduction of MIS & Strategic Management of Business

Unit 5: MIS: Concept, Definition, Role of the MIS, Impact of MIS, MIS and the user, Management as a control system.

Unit 6: MIS support to the management, Management effectiveness and MIS, Organization as system. MIS: organization effectiveness.

Unit 7: Concept of corporate planning, Essentiality of strategic planning, Development of the business strategies, Type of strategies, short-range planning, tools of planning, MIS: strategic business planning.

Block III: Development of MIS & Developing Business/IT Strategies/IT Solutions

Unit 8: Development of long-range plans of the MIS, Ascertaining the class of information, Determining the information requirement.

Unit 9: Development and implementation of the MIS, Management of information quality inthe MIS, Organization for development of MIS, MIS development process model.

Unit 10: Planning fundamentals (real world cases), Organizational planning, planning for competitive advantage, (SWOT Analysis), Business models and planning. Business/IT planning,

Unit 11: Identifying business/IT strategies, Implementation Challenges, Change management., Developing business systems, (real world case), SDLC, prototyping, System development process, implementing business system.

Block IV: Business Process Re-Engineering & Technology of Information System

Unit 12: Introduction, Business process, process model of the organization, value stream model of the organization.

Unit 13: What delay the business process, relevance of information technology, MIS and BPR.

Unit 14: Introduction, Data processing, Transaction processing, Application processing, information system processing.

Unit 15: TQM of information systems, Human factors & user interface, Strategic nature of IT decision, MIS choice of information technology.

Block V: Decision Making and DSS, Data resource Management

Unit 16: Decision making concepts; decision making process, decision-making by analytical modeling, Behavioral concepts in decision making.

Unit 17: Organizational decision-making, Decision structure, DSS components, Management reporting alternatives.

Unit 18: Managing data sources, Foundation concepts of data, types of databases, traditional file processing, DBMS approach, Database structure, Database development.

Unit 19: Enterprise business system – Introduction, cross-functional enterprise applications, real world case, Functional business system, - Introduction, marketing systems, sales force automation, CIM, HRM.

Unit 20: Online accounting system, Customer relationship management, ERP, Supply chain management, Electronic commerce fundamentals, e-Commerce applications and Issues, (real world cases).

Books Recommended/Suggested Reading:

1. Management Information Systems: Managing the Digital Firm" by Kenneth C. Laudon and Jane P. Laudon
2. Information Systems for Managers: Texts and Cases" by Gabe Piccoli

3. Essentials of Management Information Systems" by Kenneth C. Laudon and Jane P. Laudon
4. Information Systems: A Manager's Guide to Harnessing Technology" by John Gallaugher
5. Business Information Systems by Paul Bocij, Andrew Greasley, and Simon Hickie.

Course Code: CSM - 6214 **Credit: 3**
Course Name: Design & Analysis of Algorithm

Course objectives:

- To introduce students to various algorithm design techniques, including greedy algorithms, divide and conquer, dynamic programming, and backtracking, enabling them to develop efficient algorithms for solving computational problems.
- To teach students how to analyse the efficiency of algorithms in terms of time complexity and space complexity, using mathematical tools such as big-O notation, and to understand the implications of algorithmic efficiency on performance and scalability.
- To familiarize students with essential data structures such as arrays, linked lists, stacks, queues, trees, and graphs, and their applications in algorithm design and implementation.
- To develop students' problem-solving skills by providing them with opportunities to apply algorithm design techniques to solve a variety of computational problems, including sorting, searching, graph algorithms, and optimization problems.
- To explore advanced algorithmic paradigms such as approximation algorithms, randomized algorithms, and parallel algorithms, and their use in solving complex computational problems efficiently.

Course Outcomes (CO's):

On successful completion of this practical lab session, students will be able to:
1. Proficiency in designing efficient algorithms for solving complex computational problems.
2. Mastery in analyzing algorithm performance and complexity using mathematical techniques.
3. Enhanced problem-solving skills through the application of algorithmic design paradigms.
4. Ability to evaluate and compare different algorithmic approaches for optimizing computational tasks.
5. Understanding of algorithmic principles to address real-world challenges in computer science and engineering.

Block – I: Introduction to Algorithms
Unit -1: Basics of an Algorithm and its properties: Introduction, Objective, Example of an Algorithm, Basics building blocks of Algorithms, A survey of common running time, Analysis & Complexity of Algorithm, Types of problems, Problem Solving Techniques, Deterministic and Stochastic Algorithms.
Unit 2: Some pre-requisites and Asymptotic Bounds: Introduction, Objectives, Some Useful Mathematical Functions & Notations Functions & Notations Modular Arithmetic/Mod Function, Mathematical Expectation, Principle of Mathematical Induction, Concept of Efficiency of an Algorithm.
Unit 3: Analysis of Simple Algorithm: Introduction, Objectives, Complexity Analysis of Algorithms Euclid Algorithm for GCD Polynomial Evaluation Algorithm Exponent Evaluation Sorting Algorithm, Analysis of Non-Recursive Control Structures Sequencing for Construct While and Repeat Constructs Recursive Constructs.
Unit 4: Solving Recurrences: Introduction, Objective, Substitution Methods, Iteration Methods, Recursive Tree Methods, Master Methods.

Block – II: Design Techniques-I

Unit 5: Greedy Technique, Some Examples to understand Greedy Techniques, Formalization of Greedy Techniques.
Unit 6: An overview of local and global optima, Fractional Knapsack problem, Huffman Codes, A task scheduling algorithm.
Unit 7: Divide & Conquer Technique, General Issues in Divide and Conquer Technique, Binary Search.
Unit 8: Algorithm, Sorting Algorithm, Merge Sort, Quick Sort, Matrix Multiplication Algorithm.
Unit 9: Graph Algorithm – I: Basic Definition and terminologies, Graph Representation, Adjacency Matrix, Adjacency List.
Unit 10: Graph Traversal Algorithms, Depth First Search, Breadth First Search, Topological Sort, Strongly Connected Components.

Block – III: Design Techniques – II
Unit 11: Graph Algorithms – II: Minimum Cost Spanning Tree problems, Kruskal's Algorithm, Prim's Algorithm, Single Source Shortest Path Problems.
Unit 12: Bellman Ford Algorithm Dijkstra's Algorithm, Maximum Bipartite Matching Problem.
Unit 13: Dynamic Programming Technique, The Principle of Optimality, Chained Matrix Multiplication, Matrix Multiplication Using Dynamic Programming.
Unit 14: Optimal binary search trees problems, Binomial coefficient computation, Floyd Warshall algorithm.
Unit 15: String Matching Techniques, The naïve String-Matching Algorithm, The Rabin Karp Algorithm, Knuth –Morris Pratt Algorithm.

Block – IV: NP- Completeness and Approximation Algorithm
Unit 16: NP-Completeness, Concepts of Class-P, NP Completeness, NP-Hard, Unsolvable problems, Polynomial-time, Polynomial-time Reductions, Class P with Examples, Knapsack and TSP problems.
Unit 17: NP-Completeness and NP- hard Problems, Polynomial Time verification, Techniques to show NP- Hardness, NP-Complete problems and P Vs NP problems.
Unit 18: Handling Intractability, Approximation algorithms for Vertex Cover problem and minimizing make span as parallel machines (Graham's algorithm), Parameterized algorithm for Vertex Cover problem, Meta-heuristic Algorithms.

Books Recommended/Suggested Reading:

1. Horowitz Sahani, "Fundamentals of Computer Algorithms", Golgotia
2. Coremen Leiserson etal, "Introduction to Algorithms", PHI
3. Brassard Bratley, "Fundamental of Algorithms", PHI
4. M T Goodrich etal, "Algorithms Design", John Wiley
5. A V Aho etal, "The Design and analysis of Algorithms", Pearson Education.

Course Code: CSM - 6215 **Credit: 3**

Course Name: Data Structure using C++

Course objectives:

- To introduce students to fundamental data structures such as arrays, linked lists, stacks, queues, trees, and graphs, and their applications in solving computational problems efficiently.
- To teach students how to implement data structures using object-oriented programming principles in C++, including encapsulation, inheritance, polymorphism, and abstraction.
- To enable students to analyse the time and space complexity of algorithms and data structures, using mathematical tools such as big-O notation, and to understand the implications of algorithmic efficiency on performance and scalability.
- To develop students' problem-solving skills by providing them with opportunities to apply data structures and algorithms to solve a variety of computational problems, including sorting, searching, graph algorithms, and dynamic programming.
- To provide students with practical experience in implementing and using data structures and algorithms through hands-on lab sessions, enabling them to gain proficiency in programming and problem-solving using C++.

Course Outcomes (CO's):

On successful completion of this practical lab session, students will be able to:

1. Students will demonstrate proficiency in implementing and manipulating fundamental data structures such as arrays, linked lists, stacks, queues, trees, and graphs using C++.
2. They will develop competence in analyzing problem requirements and selecting appropriate data structures to efficiently organize and manage data.
3. Through practical application, students will master file handling operations, including reading from and writing to files, and navigating directories using C++.
4. They will exhibit the ability to design and implement efficient algorithms for various data and file structure manipulation tasks.
5. The course will empower students to apply their knowledge and skills in solving real-world problems requiring the manipulation of data and files using C++.

Block I: Introduction to Algorithms and Data Structures

Unit 1: Analysis of Algorithms: Mathematical Background, Process of Analysis, Calculation of Storage Complexity, Calculation of Run Time Complexity.

Unit 2: Arrays, Pointers and Structures, Arrays and Pointers, Sparse Matrices, Structures, Polynomials, Representation of Arrays, Row Major Representation, Column Major Representation, Applications.

Unit 3: Lists, Abstract Data Type-List, Array Implementation of Lists, Linked Lists-Implementation, Doubly Linked Lists-Implementation, Circularly Linked Lists-Implementation, Skip lists, Applications.

Block II: Stacks, Queues and Trees

Unit 4: Stacks, Abstract Data Type-Stack, Implementation of Stack, Implementation of Stack using Arrays, Implementation of Stack using Linked Lists, Algorithmic Implementation of Multiple Stacks, Applications.
Unit 5: Queues, Abstract Data Type-Queue, Implementation of Queue, Array Implementation, Linked List Implementation, Implementation of Multiple Queues, Implementation of Circular Queues, Array Implementation.
Unit 6: Linked List Implementation of a circular queue, Priority Queues, Implementation of DEQUEUE, Array Implementation of a dequeue, Linked List Implementation of a dequeue.
Unit 7: Trees: Abstract Data Type-Tree, Implementation of Tree, Tree Traversals, Binary Trees, Implementation of Binary Tree.
Unit 8: Binary Tree Traversals, Recursive Implementation of Binary Tree Traversals, Non Recursive Implementations of Binary Tree Traversals, Applications.

Block III: Graph Algorithms and Searching Techniques
Unit 9: Advanced Trees: Binary Search Trees, Traversing a Binary Search Trees, Insertion of a node into a Binary Search Tree, Deletion of a node from a Binary Search Tree, AVL Trees: Insertion of a node into an AVL Tree, Deletion of a node from and AVL Tree, AVL tree rotations, Applications of AVL Trees.
Unit 10: B-Trees: Operations on B-Trees, Applications of B-Trees, Splay Trees, Splaying steps, Splaying Algorithm, Red-Black trees, Properties of a Red-Black tree, AA-Trees.
Unit 11: Graphs: Definitions, Shortest Path Algorithms, Dijkstra's Algorithm, Graphs with Negative Edge costs, Acyclic Graphs, All Pairs Shortest Paths Algorithm.
Unit 12: Minimum cost Spanning Trees, Kruskal's Algorithm, Prims's Algorithm, Applications, Breadth First Search, Depth First Search, Finding Strongly Connected Components.
Unit 13: Searching and Sorting Techniques: Linear Search, Binary Search, Applications, Internal Sorting, Insertion Sort, Bubble Sort, Quick Sort, 2-way Merge Sot, Heap Sort, Sorting on Several Keys, External Sorting Algorithms.

Block IV: File Structures and Advanced Data Structures
Unit 14: Hashing: Introduction, Index Mapping, Collision Handling, Double Hashing, Load Factor and Rehashing.
Unit 15: Advanced Data Structures: Scapegoat Trees, Tries, Binary Tries, X-Fast Tries, Y-Fast Tries.
Unit 16: File Structures: Terminology, File Organisation, Sequential Files, Structure, Operations, Disadvantages, Areas of use, Direct File Organisation, Indexed Sequential File Organisation.

Books Recommended/Suggested Reading:

1. Data Structures and Algorithm Analysis in C++" by Mark Allen Weiss
2. Data Structures Using C++" by D. S. Malik
3. Data Structures and Algorithms in C++" by Adam Drozdek
4. Data Structures and Other Objects Using C++" by Michael Main and Walter Savitch
5. Algorithms in C++ Part 1-4: Fundamentals, Data Structures, Sorting, Searching" by Robert Sedgewick
6. Effective C++: 55 Specific Ways to Improve Your Programs and Designs" by Scott Meyers.

Course Code: CSM - 6251 **Credit: 2**

Course Name: DAA and Web Design Lab

Course objectives:

- Implementing and analysing various algorithmic techniques such as greedy algorithms, dynamic programming, divide and conquer, and backtracking.
- Analysing the time and space complexity of algorithms using mathematical tools like big-O notation and understanding their implications on algorithmic efficiency.
- Developing practical skills in web development by designing and building interactive and dynamic web pages using HTML, CSS, and JavaScript.
- Exploring advanced concepts in web design, such as responsive design, web accessibility, and user experience (UX) design, to create user-friendly and visually appealing websites.
- Integrating client-side scripting with server-side scripting languages such as PHP or Node.js to create dynamic and database-driven web applications.

Course Outcomes (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in designing and analyzing algorithms for solving computational problems in DAA Lab.
2. Mastery in implementing and evaluating various algorithmic approaches for optimization in DAA Lab.
3. Skills in creating dynamic and user-friendly websites using HTML, CSS, and JavaScript in Web Design Lab.
4. Ability to design responsive web interfaces and incorporate multimedia elements for enhanced user experience in Web Design Lab.
5. Understanding of web design principles and techniques to develop functional and visually appealing websites in Web Design Lab.

Lab Sessions:

- There will be practical sessions of which 10 sessions will be on DAA and 10 sessions will be on Web Designing.
- The practice problems for all 20 sessions will be listed session-wise in the lab manual.

Books Recommended/Suggested Reading:

For Design and Analysis of Algorithms (DAA):

1. Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.
2. Algorithm Design" by Jon Kleinberg and Éva Tardos
3. The Algorithm Design Manual" by Steven S. Skiena

For Web Programming:

4. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Niederst Robbins
5. "JavaScript and JQuery: Interactive Front-End Web Development" by Jon Duckett

Course Code: CSM - 6252 **Credit: 2**

Course Name: Data Structure using C++

Objective:

- To familiarize students with the basic syntax, concepts, and features of the C++ programming language, including variables, data types, control structures, functions, and classes.
- To provide students with practical experience in writing, compiling, and executing C++ programs, covering a variety of programming constructs and problem-solving techniques.
- To reinforce the concepts of object-oriented programming (OOP) through hands-on exercises, including class and object creation, inheritance, polymorphism, and encapsulation.
- To teach students how to debug and troubleshoot C++ programs effectively, including techniques for identifying and fixing common errors and logical flaws in code.
- To enable students to develop applications and projects using C++ programming, applying their knowledge of programming fundamentals and OOP concepts to solve real-world problems and implement software solutions.

Course Outcomes (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in implementing object-oriented programming concepts and practices in C++ Lab.
2. Mastery in developing efficient algorithms and data structures using C++ programming language in C++ Lab.
3. Skills in debugging and troubleshooting C++ code to identify and resolve programming errors in C++ Lab.
4. Ability to create robust and scalable software solutions through hands-on coding exercises and projects in C++ Lab.
5. Understanding of memory management, file handling, and modular programming in C++ for building comprehensive software applications in C++ Lab.

C++ Programming Lab:
Syllabus and Sessions Allocation: (10 Sessions)

Session 1: Basics of C++, data type, I/O, Control Structures etc.

Session 2: Class and Objects, function calling.

Session 3: Constructor and Destructor.

Session 4: Inheritance.

Session 5: Operator Overloading.

Session 6: Polymorphism.

Session 7: Template class and function.

Session 8: I/O and streaming.

Session 9: Exception Handling

Session10: STL.

Books Recommended/Suggested Reading:

1. Data Structures and Algorithm Analysis in C++" by Mark Allen Weiss
2. Data Structures Using C++" by D. S. Malik
3. Data Structures and Algorithms in C++" by Adam Drozdek
4. Data Structures and Other Objects Using C++" by Michael Main and Walter Savitch
5. Algorithms in C++ Part 1-4: Fundamentals, Data Structures, Sorting, Searching" by Robert Sedgewick
6. Effective C++: 55 Specific Ways to Improve Your Programs and Designs" by Scott Meyers.

SEMESTER – III

Course Code: CSM - 7111 **Credit: 3**

Course Name: Artificial Intelligence and Machine Learning

Course objectives:

- To provide students with a foundational understanding of artificial intelligence (AI) and machine learning (ML), including key concepts, algorithms, and applications in various domains.
- To introduce students to various machine learning techniques, including supervised learning, unsupervised learning, and reinforcement learning, and their applications in tasks such as classification, regression, clustering, and reinforcement learning.
- To familiarize students with deep learning techniques and neural networks, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and generative adversarial networks (GANs), and their applications in image recognition, natural language processing, and other domains.
- To teach students how to preprocess and prepare data for machine learning models, including techniques such as data cleaning, feature selection, transformation, and normalization.
- To provide students with practical experience in implementing machine learning algorithms and models using popular libraries and frameworks such as scikit-learn, TensorFlow, and PyTorch, and applying them to real-world datasets and problems.

Course Outcomes (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in implementing AI and ML algorithms to solve complex real-world problems.
2. Mastery in analyzing and interpreting large datasets using machine learning techniques.
3. Skills in developing intelligent systems capable of learning from data and making informed decisions autonomously.
4. Ability to design and deploy AI models for tasks such as image recognition, natural language processing, and predictive analytics.
5. Understanding of ethical considerations and societal implications of AI and ML applications in diverse domains.

Block – I: Artificial Intelligence - Introduction:

Unit-1 Introduction to Artificial Intelligence - What is AI? Examples of AI systems, Approaches to AI, Brief history of AI, Comparison Between Artificial intelligence, Machine Learning, and Deep Learning, Intelligent Agent: stimulus response agents. components of intelligence.

Unit-2 Problem Solving using Search – Single agent search: Introduction to State Space Search, Statement of Search problems: state space graphs, Searching explicit state spaces. Feature based state spaces. Problem types, examples (puzzle problem, n-queen, the road map) Two agent search: Adversarial search: Two agent games (alpha-beta pruning). Min Max Search.

Unit-3 Uninformed and Informed Search – Uninformed Search: Formulating the state space, iterative deepening, bidirectional search. Informed Search Strategies: Using evaluation functions. A & AO, admissibility of A, Iterative deepening A, recursive best first search.

Unit-4 Predicate and Propositional Logic – Propositional logic, syntax, semantics, semantic rules, terminology - validity, satisfiability. interpretation, entailment, proof systems. Propositional Logic inference rules, natural deduction, propositional resolution.

Block – II: Artificial Intelligence - Knowledge Representation:

Unit-5 First Order Logic - First Order Logic: Motivation, Syntax, Interpretations, semantics of quantifiers, Entailment in FOL, Interpretation, Inference in FOL: First Order resolution. Conversion to clausal form. Unification. Most general unifier. Resolution with variables Proving validity.

Unit-6 Rule based Systems and other formalism - Rule Based Systems: Forward chaining.

Backward chaining. Conflict resolution. Semantic nets, Frames, Scripts.

Unit-7 Probabilistic Reasoning Reasoning with uncertain information Review of Probability Theory, Introduction to Bayesian Theory, Bayes' Networks, Probabilistic Inference, Basic idea of inferencing with Bayes networks. Other paradigms of uncertain reasoning. Dempster-Scheffer Theory.

Unit-8 Fuzzy and Rough Set Fuzzy Reasoning Introduction to Fuzzy sets, Fuzzy set representation, Fuzzy inferences, Rough Set Theory.

Block – III: Machine Learning - I:

Unit-9 Introduction to Machine Learning Methods – Introduction to Machine Learning, Techniques of Machine Learning, Reinforcement Learning and algorithms, Deep Learning and its Algorithms, Ensemble Methods.

Unit-10 Classification – Understanding of Supervised Learning, Introduction to Classification, Classification Algorithms: Naïve Bayes, K-NN, Decision Trees, Logistic Regression, Support Vector Machines.

Unit-11 Regression – Introduction to Regression, Regression algorithm Linear Regression and Polynomial Regression, Support Vector Regression.

Unit-12 Neural Networks and Deep Learning: Overview of Artificial Neural Networks, Multilayer Feedforward Neural networks with Sigmoid activation functions; Back propagation Algorithm; Representational abilities of feed forward networks, Feed forward networks for Classification and Regression, Deep Learning.

Block – IV: Machine Learning - II:

Unit-13 Feature selection and Extraction: Introduction to Feature Selection and Extraction, Dimensionality Reduction, Principal Component Analysis, Linear Discriminant Analysis, Singular Value Decomposition.

Unit-14 Association Rules – Introduction to Pattern search and its algorithms: Apriori Algorithms. and its variants, FP Tree Growth, Pincer Search.

Unit-15 Clustering – Introduction to Clustering, Types of Clustering, Partition Based, Hierarchical Based, Density Based Clustering Techniques, Clustering algorithms: K Means, Agglomerative and Divisive, DBSCAN, Introduction to Fuzzy Clustering.

Unit – 16 Machine Learning Programming using Python Implementations of various algorithms of this course.

Books Recommended/Suggested Reading:

1. Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig.
2. Pattern Recognition and Machine Learning" by Christopher M. Bishop.

3. Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy.
4. Deep Learning by Ian Goodfellow, Yoshua Bengio, and Aaron Courville.
5. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron.

Course Code: CSM – 7112 **Credit: 3**
Course Name: Data Science using R Programming

Course Objectives:

- To introduce the fundamentals of data science and its applications using R.
- To develop skills in data manipulation, visualization, and statistical analysis.
- To implement machine learning algorithms using R programming.
- To enhance problem-solving abilities through real-world data analysis.
- To apply data science techniques for predictive modeling and decision-making.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Understand core concepts of data science and R programming.
2. Perform data manipulation, cleaning, and visualization using R.
3. Implement statistical methods and machine learning models in R.
4. Analyze and interpret real-world datasets for insights and predictions.
5. Develop data-driven solutions for business and research applications.

Block – I: Fundamentals of Data Science

Unit 1: Introduction to Data Science

- Definition and Scope of Data Science
- Types of Data Analysis (Descriptive, Exploratory, Inferential, Causal, Predictive)
- Common Mistakes in Data Analysis (Correlation vs. Causation, Simpson's Paradox, Data Dredging)
- Applications and Life Cycle of Data Science

Unit 2: Probability and Statistics for Data Science

- Basic Concepts: Probability, Dependence & Independence, Conditional Probability
- Bayes' Theorem, Random Variables, Probability Distributions
- Normal Distribution, Central Limit Theorem
- Hypothesis Testing, Confidence Intervals

Unit 3: Data Preparation for Analysis

- Data Preprocessing Techniques
- Data Cleaning and Transformation
- Feature Selection and Extraction
- Data Integration and Knowledge Discovery

Unit 4: Data Visualization and Interpretation

- Types of Plots: Histograms, Boxplots, Scatter Plots
- Regression-related Visualizations
- Data Interpretation with Real-World Examples

Block – II: Big Data and Its Management
Unit 5: Big Data Architecture

- Characteristics of Big Data (Volume, Variety, Velocity, Veracity)
- Structured vs. Semi-Structured and Unstructured Data
- Big Data vs. Data Warehousing
- Hadoop Architecture: HDFS, YARN, MapReduce

Unit 6: Programming with MapReduce
- Fundamentals of MapReduce Processing
- Loading Data into HDFS
- Execution of Map and Reduce Phases
- Algorithms Using MapReduce (Word Counting, Matrix-Vector Multiplication)

Unit 7: Big Data Tools and Technologies
- Apache Spark: Architecture and Functionality
- Data Processing with Hive and HBase
- Overview of Other Big Data Tools

Unit 8: NoSQL Databases
- Column-Based, Graph-Based, Key-Value Pair, and Document-Based Databases
- Use Cases and Applications

Block – III: Big Data Analysis
Unit 9: Mining Big Data

- Finding Similar Items and Jaccard Similarity
- Collaborative Filtering and Similarity Measures
- Euclidean Distance in Data Analysis

Unit 10: Data Stream Processing
- Data Stream Models and Management
- Query Processing in Data Streams
- Challenges and Sampling Techniques
- Filtering and Counting Unique Elements in Streams

Unit 11: Link Analysis & Page Ranking

- Concepts and Mechanisms of Page Rank
- Link Spam Detection and Topic-Sensitive Page Rank
- Hubs and Authorities in Web Search

Unit 12: Web and Social Network Analysis

- Introduction to Web Analytics and Online Advertising
- Recommendation Systems and Utility Matrix
- Social Network Graphs and Distance Measures
- Clustering Techniques in Social Media Analysis

Block – IV: Programming for Data Analysis

Unit 13: Introduction to R Programming

- R Environment and Setup
- Data Types, Variables, Operators
- Decision Making, Loops, and Functions
- Data Structures in R: Vectors, Lists, Data Frames, Matrices, Arrays

Unit 14: Data Interfacing and Visualization in R

- Reading/Writing Data: CSV, Excel, XML, JSON, Databases
- Data Cleaning and Processing in R
- Visualization Techniques: Bar Charts, Box Plots, Histograms, Line Charts, Scatter Plots

Unit 15: Statistical Data Analysis in R

- Chi-square Test, Linear and Multiple Regression
- Logistic Regression and Time Series Analysis

Unit 16: Advanced Data Analysis in R

- Decision Trees and Random Forest
- Classification, Clustering, and Association Rule Mining

Books Recommended/Suggested Reading:

1. "Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking" by Foster Provost and Tom Fawcett
2. "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython" by Wes McKinney
3. "Big Data: Principles and Best Practices of Scalable Real-Time Data Systems" by Nathan Marz and James Warren
4. "Data Science from Scratch: First Principles with Python" by Joel Grus
5. "Mining of Massive Datasets" by Jure Leskovec, Anand Rajaraman, and Jeffrey D. Ullman

Course Code: CSM – 7113 **Credit: 3**
Course Name: OOP's Technologies and Java Programming

Course Objectives:

- To introduce students to the core principles of OOP, including encapsulation, inheritance, polymorphism, and abstraction, and their significance in software development.
- To familiarize students with the syntax, data types, control structures, and basic concepts of the Java programming language, enabling them to write simple Java programs and understand object-oriented concepts in the context of Java.
- To teach students advanced features of Java programming, including generics, collections, exception handling, multithreading, and input/output operations, enabling them to develop robust and efficient Java applications.
- To guide students in applying OOP principles and design patterns to solve real-world problems, including modeling software systems, designing classes and interfaces, and implementing reusable and maintainable code.
- To introduce students to popular Java technologies and frameworks, such as Java EE (Enterprise Edition), Spring Framework, Hibernate, and JavaFX, and their applications in developing web applications, enterprise systems, and desktop applications.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in Java syntax, object-oriented principles, and programming constructs.
2. Mastery in developing Java applications for various platforms and environments.
3. Skills in implementing data structures, algorithms, and design patterns in Java programming.
4. Ability to debug, test, and troubleshoot Java code for robust and efficient software development.
5. Understanding of Java's role in modern software development and its applications in enterprise systems and web development.

Block–I: Object Oriented Technology and Java

Unit 1: Object Oriented Methodology-1 Paradigms of Programming Languages, Evolution of OO Methodology, Basic Concepts of OO Approach.

Unit 2: Comparison of Object Oriented and Procedure Oriented Approaches, Benefits of OOPs, Introduction to Common OO Language, Applications of OOPs.

Unit 3: Object Oriented Methodology-2 Classes and Objects, Abstraction and Encapsulation, Inheritance, Method Overriding and Polymorphism.

Unit 4: Java Language Basics Introduction to Java, Basic Features, Java Virtual Machine Concepts, A Simple Java Program, Primitive Data Type And Variables, Java Keywords, Integer and Floating Point Data Type, Character and Boolean Types, Declaring and Initialization Variables, Java Operators.

Unit 5: Expressions, Statements and Arrays Expressions, Statements, Control Statements, Selection Statements, Iterative Statements, Jump Statements, Arrays.

Block–II: Object Oriented Concepts and Exceptions Handling

Unit 6: Class and Objects Class Fundamentals, Creating objects, Assigning object reference variables, Introducing Methods, Static methods, Constructors, Overloading constructors.

Unit 7: Keyword, Using Objects as Parameters, Argument passing, Returning objects, Method Overloading, Garbage Collection, The Finalize ( ) Method.
Unit 8: Inheritance and Polymorphism Inheritance Basics, Access Control, Multilevel Inheritance, Method Overriding, Abstract Classes, Polymorphism, Final Keyword.
Unit 9: Packages and Interfaces Package, Defining Package, CLASSPATH, Package naming, Accessibility of Packages, Using Package Members, Interfaces, Implementing Interfaces, Interface and Abstract Classes, Extends and Implements Together.
Unit 10: Exceptions Handling Exception, Handling of Exception, Using try-catch, Catching Multiple Exceptions, Using finally clause, Types of Exceptions, Throwing Exceptions, Writing Exception Subclasses.

Block–III: Multithreading, I/O and String Handling
Unit 11: Multithreaded Programming Multithreading: An Introduction, The Main Thread, Java Thread Model, Thread Priorities, Synchronization in Java, Interthread Communication.
Unit 12: I/O in Java I/O Basics, Streams and Stream Classes, Byte Stream Classes, Character Stream Classes, The Predefined Streams, Reading from, and Writing to, Console, Reading and Writing Files, The Transient and Volatile Modifiers, Using Instance of Native Methods.
Unit 13: Strings and Characters Fundamentals of Characters and Strings, The String Class, String Operations, Data Conversion using Value Of ( ) Methods, String Buffer Class and Methods.
Unit 14: Exploring Java I/O Java I/O Classes and Interfaces, I/O Stream Classes, Input and Output Stream, Input Stream and Output Stream Hierarchy, Text Streams, Stream Tokenizer, Serialization, Buffered Stream, Print Stream, Random Access File.

Block–IV: Applets Programming and Advance Java Concepts
Unit 15: Applets: The Applet Class, Applet Architecture, An Applet Skeleton: Initialization and Termination, Handling Events, HTML Applet Tag.
Unit 16: Graphics and User Interfaces Graphics Contexts and Graphics Objects, Color Control, Fonts, Coordinate System, User Interface Components, Building User Interface with AWT, Swing-based GUI, Layouts and Layout Manager, Container.
Unit 17: Networking Features Socket Overview, Reserved Parts and Proxy Servers, Internet Addressing: Domain Naming Services (DNS), JAVA and the net: URL, TCP/IP Sockets.
Unit 18: Advance Java Java Database Connectivity, Establishing A Connection, Transactions with Database, An Overview of RMI Applications, Remote Classes and Interfaces, RMI Architecture, Java Beans.

Books Recommended/Suggested Reading:

For Object-Oriented Programming (OOP) Technologies:
1. Object-Oriented Analysis and Design with Applications" by Grady Booch.
2. Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides.
3. Object-Oriented Programming in C++" by Robert Lafore.
For Java Programming:
4. Effective Java" by Joshua Bloch
5. Java: The Complete Reference" by Herbert Schildt.

Course Code: CSM – 7114 **Credit: 3**
Course Name: Advanced Database Management System

Course Objectives:

- Understand advanced database concepts, including normalization, indexing, and query optimization.
- Explore distributed databases, NoSQL databases, and modern data storage techniques.
- Implement transaction management, concurrency control, and recovery mechanisms.
- Analyze database security models and access control techniques.
- Develop scalable database solutions using emerging database technologies.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Apply advanced database design principles for efficient data management.
2. Optimize queries and indexing techniques for improved database performance.
3. Implement distributed and NoSQL databases for handling large-scale data.
4. Ensure data integrity, consistency, and security in multi-user environments.
5. Design and develop real-world database applications with advanced functionalities.

Block – I: Advanced Database Concepts

Unit 1: Introduction to ADBMS - Need for Advanced Databases, Comparison with Traditional DBMS, Database Architectures.

Unit 2: Data Models and Query Processing - Enhanced ER Model, Object-Oriented Databases, XML Databases, JSON Databases.

Unit 3: Normalization and Indexing - Higher Normal Forms (BCNF, 4NF, 5NF), Indexing Techniques (B-Trees, Hashing), Query Optimization.

Unit 4: Query Processing and Optimization - Cost Estimation, Heuristic Optimization, Join Algorithms, Query Caching.

Block – II: Transaction Management and Concurrency Control

Unit 5: Transaction Processing - ACID Properties, Serializability, Schedule Classification, Two-Phase Commit Protocol.

Unit 6: Concurrency Control - Locking Mechanisms, Timestamp Ordering, Deadlock Detection and Prevention.

Unit 7: Recovery Mechanisms - Log-Based Recovery, Checkpointing, Shadow Paging, ARIES Algorithm.

Unit 8: Database Security - Access Control, Authentication, Authorization, SQL Injection, Role-Based Security.

Block – III: Distributed and NoSQL Databases

Unit 9: Distributed Databases - Concepts, Architecture, Data Fragmentation, Replication, and Allocation.

Unit 10: Query Processing in Distributed Databases - Distributed Query Execution, Distributed Transactions, CAP Theorem.

Unit 11: NoSQL Databases - Key-Value Stores, Document-Oriented Databases, Column-Family Stores, Graph Databases.

Unit 12: Big Data and Modern Database Trends - Hadoop, HDFS, MapReduce, Cloud Databases, Data Warehousing.

Block – IV: Advanced Topics and Applications

Unit 13: Object-Relational Databases - Features, ORDBMS Architecture, Case Studies.

Unit 14: XML and JSON Databases - Data Storage and Retrieval, XQuery, XPath, JSON Parsing.

Unit 15: Data Mining and Warehousing - Data Preprocessing, OLAP, Data Cube Computation, Association Rule Mining.

Unit 16: Emerging Database Technologies - Blockchain Databases, In-Memory Databases, Graph Databases in AI/ML Applications.

Books Recommended/Suggested Reading:

1. "Database System Concepts" – Abraham Silberschatz, Henry F. Korth, S. Sudarshan (McGraw-Hill)
2. "Fundamentals of Database Systems" – Ramez Elmasri, Shamkant B. Navathe (Pearson Education)
3. "Principles of Distributed Database Systems" – M. Tamer Özsu, Patrick Valduriez (Springer)
4. "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence" – Pramod J. Sadalage, Martin Fowler (Addison-Wesley)
5. "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling" – Ralph Kimball, Margy Ross (Wiley)

Course Code: CSM – 7151 **Credit: 2**
Course Name: Java Programming Lab

Course Objectives:

- To provide students with hands-on experience in writing, compiling, and executing Java programs, covering basic syntax, data types, control structures, and functions.
- To reinforce the concepts of object-oriented programming (OOP) through practical exercises, including class and object creation, inheritance, polymorphism, and encapsulation.
- To enable students to develop Java applications and projects, including console-based applications, graphical user interfaces (GUIs), and web applications, using Java SE (Standard Edition) and Java EE (Enterprise Edition) technologies.
- To teach students how to connect Java applications to databases using JDBC (Java Database Connectivity), enabling them to perform database operations such as querying, inserting, updating, and deleting data.
- To introduce students to advanced features of Java programming, including multithreading, networking, file handling, and exception handling, and their applications in building robust and efficient Java applications.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in implementing Java programming concepts to solve real-world problems.
2. Mastery in designing and developing Java applications through hands-on coding exercises.
3. Skills in debugging and troubleshooting Java code to ensure program correctness and efficiency.
4. Ability to apply object-oriented design principles to create modular and scalable software solutions.
5. Understanding of software development methodologies and best practices in Java programming.

Section - I: Java Programming Lab

- Programming with Java
- Path and Class Path Setting
- Example Programs
- List of Lab Assignments

List of 20 Java Programs (10 Basic & 10 Advanced)

Basic Programs

1. **Basic Java Program:** Write a Java program to print "Hello, World!" on the console.
2. **Arithmetic Operations:** Write a Java program to perform addition, subtraction, multiplication, and division using user input.
3. **Check Even or Odd:** Write a Java program to check whether a given number is even or odd.
4. **Find Largest Number:** Write a Java program to find the largest of three numbers using if-else statements.
5. **Factorial Calculation:** Write a Java program to calculate the factorial of a given number using recursion.

6. **Fibonacci Series:** Write a Java program to print the Fibonacci series up to n terms using iteration.
7. **Palindrome Number:** Write a Java program to check whether a given number is a palindrome.
8. **Reverse a String:** Write a Java program to reverse a given string using loops.
9. **Armstrong Number:** Write a Java program to check whether a given number is an Armstrong number.
10. **Array Operations:** Write a Java program to find the sum and average of elements in an array.

Advanced Programs

11. **Class and Object Concept:** Write a Java program to create a class Student with attributes like name, rollNumber, and marks, and display the student details.
12. **Constructor Overloading:** Write a Java program to demonstrate constructor overloading with different types of constructors.
13. **Method Overriding & Inheritance:** Write a Java program to demonstrate method overriding and inheritance using a Base and Derived class.
14. **Multithreading Implementation:** Write a Java program to implement multithreading using the Thread class and Runnable interface.
15. **Exception Handling:** Write a Java program to demonstrate the use of try, catch, finally, and throws for handling exceptions.
16. **File Handling:** Write a Java program to read a file and display its contents on the console.
17. **Interface Implementation:** Write a Java program to implement multiple interfaces in a single class.
18. **Database Connectivity:** Write a Java program to connect to a MySQL database using JDBC and perform CRUD operations.
19. **Swing GUI Application:** Write a Java program to create a simple GUI-based calculator using JFrame, JButton, and JTextField.
20. **Collections Framework:** Write a Java program to demonstrate the use of ArrayList, HashMap, and LinkedList with data insertion, deletion, and iteration.

Recommended/Suggested Readings:

1. "Java: The Complete Reference" – Herbert Schildt (McGraw-Hill)
2. "Core Java Volume I – Fundamentals" – Cay S. Horstmann (Pearson Education)
3. "Head First Java" – Kathy Sierra, Bert Bates (O'Reilly Media)
4. "Java Programming for Beginners" – Mark Lassoff (BPB Publications)
5. "Effective Java" – Joshua Bloch (Addison-Wesley)

Course Code: CSM – 7152 **Credit: 2**
Course Name: Data Science using R Programming Lab

Course Objectives:

- Develop proficiency in R programming for data manipulation, analysis, and visualization.
- Understand statistical methods and machine learning techniques using R.
- Apply data pre-processing, cleaning, and transformation techniques for real-world datasets.
- Implement data visualization techniques to interpret and communicate analytical results effectively.
- Gain hands-on experience in building predictive models and performing exploratory data analysis.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Demonstrate the ability to write and execute R scripts for data analysis tasks.
2. Utilize statistical and machine learning libraries in R for data-driven decision-making.
3. Perform data preprocessing, handling missing values, and feature engineering efficiently.
4. Create insightful visualizations to interpret complex datasets and trends.
5. Develop predictive models and evaluate their performance using appropriate R packages.

Data Science Using R Programming

Basic Programs (Fundamentals of R and Data Handling)

1. Write a program to perform basic arithmetic operations in R.
2. Write a program to demonstrate the use of vectors, matrices, lists, and data frames in R.
3. Write a program to import and export data from CSV, Excel, and JSON files.
4. Write a program to implement control structures (if-else, loops) in R.
5. Write a program to perform basic statistical functions (mean, median, variance, standard deviation, correlation).
6. Write a program to visualize data using bar charts, histograms, and scatter plots.
7. Write a program to perform basic data preprocessing, including handling missing values and outliers.
8. Write a program to apply string operations (concatenation, substring, replace, split) in R.
9. Write a program to perform linear regression using the lm() function in R.
10. Write a program to create a simple function in R and use it to perform data manipulation.

Advanced Programs (Machine Learning & Data Science Techniques)

11. Write a program to implement multiple linear regression in R.
12. Write a program to implement logistic regression for binary classification.
13. Write a program to perform K-Means clustering on a dataset.

14. Write a program to implement Decision Tree classification using R.
15. Write a program to apply Principal Component Analysis (PCA) for dimensionality reduction.
16. Write a program to perform Association Rule Mining using the Apriori algorithm.
17. Write a program to implement a Naive Bayes classifier on a dataset.
18. Write a program to perform Time Series Forecasting using the ARIMA model in R.
19. Write a program to implement Random Forest classification and evaluate model accuracy.
20. Write a program to perform sentiment analysis on textual data using R.

Recommended/Suggested Books

1. "R for Data Science" **– Hadley Wickham & Garrett Grolemund (O'Reilly Media)**
2. **"The Art of R Programming"** – Norman Matloff (No Starch Press)
3. **"Data Science with R: A Hands-On Approach"** – Naveen Kumar & R. Anitha (BPB Publications)
4. **"Machine Learning with R"** – Brett Lantz (Packt Publishing)
5. **"Hands-On Programming with R"** – Garrett Grolemund (O'Reilly Media)

Any One Course from the Following 03 (Elective Course – CSM-7115/ CSM-7116/ CSM-7117)

Course Code: CSM – 7115 Credit: 4
Course Name: Soft Computing Techniques

Course Objectives:

- To provide students with an overview of soft computing techniques, including fuzzy logic, neural networks, evolutionary algorithms, and swarm intelligence, and their applications in solving complex and uncertain problems.
- To teach students the principles and techniques of fuzzy logic, including fuzzy sets, fuzzy inference systems, fuzzy rules, and fuzzy control, and their applications in modeling and controlling nonlinear systems.
- To familiarize students with artificial neural networks (ANNs), including feed forward neural networks, recurrent neural networks, and convolutional neural networks, and their applications in pattern recognition, classification, regression, and optimization.
- To introduce students to evolutionary algorithms such as genetic algorithms, genetic programming, and differential evolution, and their applications in optimization, search, and machine learning.
- To enable students to understand swarm intelligence algorithms such as particle swarm optimization (PSO), ant colony optimization (ACO), and bee colony optimization (BCO), and their applications in optimization, routing, and scheduling problems.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in applying neural networks, fuzzy logic, and evolutionary algorithms to solve complex problems.
2. Mastery in designing and optimizing soft computing models for pattern recognition and decision-making tasks.
3. Skills in implementing swarm intelligence techniques for optimization and decision support in diverse domains.
4. Ability to analyze and interpret results from soft computing models to derive actionable insights.
5. Understanding of the principles and applications of soft computing techniques in various industries and domains.

Block–I: Introduction to Soft Computing

Unit 1: Definition, requirement, necessity and adequacy; various dialects of soft computing – Evolutionary Algorithms, Fuzzy Sets and Fuzzy Logic.

Unit 2: Artificial Neural Networks - their suitability in Searching, optimization, decision matching and pattern related problems; potential areas of applications.

Unit 3: History of Evolutionary Computing: The Appeal of Evolution, Biological Terminology, Elements of Genetic Algorithm, Genetic Algorithms and Traditional Search Methods, Applications of Genetic Algorithm.

Unit 4: Genetic Algorithm in Problem Solving: Evolving Computer Programs, Data Analysis and Prediction.

Block–II: Theoretical Foundations of Genetic Algorithm

Unit 5: Schemas and the Two-Armed Bandit Problem, Royal Roads, Exact Mathematical Models of Genetic Algorithm.

Unit 6: Implementing a Genetic Algorithm: When should a Genetic Algorithm be used , Encoding a Problem for a Genetic Algorithm, Adapting the Encoding, Selection Methods , Genetic Operator.

Block–III: Introduction to Fuzzy Set Theory

Unit 7: Introduction to fuzzy sets and fuzzy logic; difference between classical and fuzzy sets; chance vs fuzziness; limitations of fuzzy systems.

Unit 8: Typical shapes of membership functions and their usage; operations on fuzzy sets: compliment, intersection, union; combinations on operations, aggregation operation.

Unit 9: Cartesian Product; Classical Relations and Fuzzy Relations; Cardinality, operations and properties of crisp and fuzzy relations; Composition of operations, Fuzzy cartesian product.

Unit 10: The linguistic variables, Reasoning in fuzzy logic, Fuzzification and defuzzification; Mamdani and Sugano Fuzzy Inference Systems.

Block–IV: Neural Network & Learning Fundamentals

Unit 11: Overview of biological neurons; McCulloch-Pitts model, Rosenblatt's Perceptron model, difference, capabilities and limitations; Model of generic computational neuron.

Unit 12: Basic activation functions; Basic Learning laws of neurons; Single layer and multilayer architectures; Feedforward and feedback networks.

Unit 13: Learning paradigms, supervised and unsupervised learning, reinforced learning; back propagation algorithm; Radial basis neurons, Generalized Regression Neural network, Probabilistic Neural Networks.

Unit 14: Competitive learning; Self Organizing Features Map, Hopfield networks, associative memories, applications of artificial neural networks. Elasticity vs plasticity dilemma, preprocessing, post processing, early stopping.

Block–V: Evolutionary Algorithms

Unit 15: Problems suitable and not suitable for applying evolutionary algorithms; Various dialects of evolutionary Algorithms; Terminology of Genetic Algorithms; Canonical Genetic Algorithm; Common representations and related reproduction operators.

Unit 16: Premature convergence, schema theorem, minimal deceptive problem and Royal Road function; fitness function, Roulette wheel selection, Rank selection, Tournament Selection; termination criteria, survivor selection, population models; parallel implementations.

Books Recommended/Suggested Reading:

1. Artificial Neural Networks: An introduction to ANN Theory and Practice, Peteus J. Braspenning, PHI publication, 2005.
2. Fuzzy Logic: A spectrum of Theoretical and Practical issues, Paul P. Wang, pearson publication 2004.
3. An Introduction to Genetic Algorithms, Milanie Mitchell, MIT Press 1998.
4. A Genetic Algorithm Tutorial, Darrell Whitley.

5. Fuzzy Sets, Fuzzy logic, and Fuzzy Systems: Selected Papers- Lotfi Asker Zadeh, George J. Kilr, Bo yuan, 2005.
6. Foundations of Fuzzy logic and Soft Computing: 12th International Fuzzy conference proceeding, 2005.
7.N.K. Sinha & M.M Gupta (Eds), Soft Computing & Intelligent System: Theory & Applications, Academic Press, 2000.
8. S.N Sivanandam, S.N Deepa, Principles of Soft Computing, 2nd Edition, Wiley.

Course Code: CSM – 7116 **Credit: 4**

Course Name: Data Warehousing and Data Mining

Course Objectives:

- To introduce students to the concepts, architecture, and components of data warehousing, including data sources, ETL (Extract, Transform, Load) processes, data warehouses, and data marts.
- To teach students how to design and implement dimensional models for data warehouses, including star schemas, snowflake schemas, and fact-dimension tables, to facilitate efficient data storage and retrieval.
- To familiarize students with the process of extracting data from various sources, transforming it into a consistent format, and loading it into a data warehouse or data mart, using tools and techniques such as SQL, scripting languages, and ETL tools.
- To introduce students to data mining concepts and techniques, including classification, clustering, association rule mining, and anomaly detection, and their applications in extracting patterns, trends, and insights from large datasets.
- To develop students' skills in visualizing and interpreting data mining results effectively, using tools and techniques such as data visualization libraries, dashboards, and storytelling, to communicate findings and insights to stakeholders.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in designing and implementing data warehouses for efficient storage and retrieval of large-scale data.
2. Mastery in utilizing data mining algorithms and techniques to uncover hidden patterns and insights from complex datasets.
3. Skills in preprocessing and cleansing data to prepare it for analysis and mining tasks.
4. Ability to interpret and visualize mining results to derive actionable insights for decision-making.
5. Understanding of ethical considerations and privacy concerns in data warehousing and mining practices.

Block – I: Data Warehouse Fundamentals And Architecture

Unit 1: Fundamentals of Data Warehouse, Introduction to Data Warehousing, Evolution of Data Warehousing, Data Warehousing Concepts, Online Transaction Processing Systems, Differences between OLTP Systems and Data Warehouse, Characteristics of Data Warehouse.

Unit 2: Data Granularity, Metadata and Data Warehousing, Functionality of Data Warehouse, Advantages of Data Warehouse, Applications of Data Warehouse, Concerns in Data Warehouse, Types of Data Warehouses, Enterprise Data Warehouse, Operational Data Store, Data Mart.

Unit 3: Data Warehouse Architecture, Introduction to Data Warehouse Architecture, Characteristics of Data Warehouse Architecture, DW Architecture Goals, Components of Data Warehouse, Load Manager, Warehouse Manager, Query Manager, Data Mart, Building Data Marts, DW and Data Marts.

Unit 4: Issues in Building Data Marts, Co-existence of DW and Data Mart, Planning and Requirements, Planning Data Warehouse and Key Issues, Planning and Project Management in constructing Data

Warehouse, Data Warehouse Development Life Cycle, Methodologies - Top- Down, Bottom-Up and Hybrid Development Methodology.
Unit 5: Dimensional Modeling, Introduction to Dimensional Modeling and its Strengths, Identifying Facts and Dimensions, Star Schema, Pros and Cons of Star Schema, Snowflake Schema, Pros and Cons of Snowflake Schema, Aggregate Tables, Need for Building Aggregate Fact Tables, Limitations of Aggregate Fact Tables, Fact Constellation Schema, Aggregate Fact Tables and Derived Dimension Tables, Pros and Cons of Fact Constellation Schema.

Block – II: ETL, OLAP and TRENDS
Unit 6: Extract, Transform and Loading, Overview of ETL, ETL requirements and steps, Data Extraction, Extraction Methods- Logical Extraction Methods and Physical Extraction Methods. Data Transformation, Basic Tasks in Transformation, Major Data Transformation Types, Data loading; Data Loading Techniques, Data Quality.
Unit 7: Introduction to Online Analytical Processing, Need for OLAP, Characteristics of OLAP, OLAP and Multidimensional Analysis, Multidimensional Logical Data Model and its Users, Multidimensional Structure, Multidimensional Operations.
Unit 8: OLAP Functions, Data Warehouse and OLAP: Hypercube & Multi-cubes, OLAP Applications, Steps in the OLAP Creation Process, Advantages of OLAP, OLAP Architectures - MOLAP, ROLAP, HOLAP, DOLAP. Trends in Data Warehouse, Data Lakes Complex Data Marts, Cloud Data Warehousing, Real Time Data Warehousing, Data Warehousing and Hadoop, Data Warehouse Automation.

Block – III: Data Mining Fundamentals and Frequent Pattern Mining
Unit 9: Data Mining – An Introduction, Introduction, Data Mining – Kind of Data, Relational Databases, Data Warehouses, Transactional Databases, Advanced Data and Informational Systems.
Unit 10: Classification of Data Mining Systems, Applications of Data Mining, Data Mining and Data Warehousing, Data Mining Tools, Major Issues in Data Mining.
Unit 11: Data Preprocessing, Introduction, Data Preprocessing Overview, Data Cleaning, Missing Values, Noisy Data, Data Cleaning as a Process, Data Integration and Transformation, Data Integration, Data Transformation.
Unit 12: Data Reduction, Data Cube Aggregation, Attribute Subset Selection, Dimensionality Reduction, Numerosity Reduction, Discretization and Binarization, Measures of Similarity and Dissimilarity-Basics.
Mining Frequent Patterns and Associations, Problem Definition, Frequent Item Set Generation.
Unit 13: APRIORI Principle, Support and Confidence Measures, Association Rule Generation, APRIORI Algorithm: Finding Frequent Item set Using Candidate Generation, Generating Association Rules from Frequent Item set, Improving the efficiency of Apriori, Correlation Analysis, From Association Analysis to Correlation Analysis.

Block – IV: Classification, Clustering and Web Mining
Unit 14: Classification, Introduction, Classification: Problem Definition, General Approaches to solving a classification problem, Evaluation of Classifiers, Classification techniques, Decision Trees-Decision tree Construction, Methods for Expressing attribute test conditions, Measures for Selecting the Best Split.
Unit 15: Algorithm for Decision tree Induction, Bayesian Classification, Bayes' Theorem o Naive-

Bayesian Classification, Bayesian Belief Networks, Support Vector Machines, The Case when the data are linearly separable, The Case when the data are linearly inseparable.
Unit 16: Clustering: Problem Definition, Clustering Overview, Categorization of Major Clustering Methods, Partitioning Method o Hierarchical Method, Density-based Method, Grid-Based Method, Model-Based Method, Constraint-based Method, Partitioning Method.
Unit 17: K-Means Algorithm, K-Medoids Hierarchical Clustering, Agglomerative Method, Divisive Method, Key Issues in Hierarchical Clustering, Strengths and Weakness, Outlier Analysis – Outlier Detection methods.
Unit 18: Text and Web Mining: Introduction, Text Data Analysis and Information Retrieval, Dimensionality Reduction for Text, Text Mining Approaches, Web mining, Web content mining, Web structure mining, Mining Multimedia Data on the Web, Automatic Classification of Web Documents, Web usage mining.

Books Recommended/Suggested Reading:

1. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling" by Ralph Kimball and Margy Ross
2. Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei
3. Building the Data Warehouse" by W. H. Inmon
4. Data Warehousing in the Real World: A Practical Guide for Building Decision Support Systems" by Sam Anahory and Dennis Murray
5. Data Mining: Practical Machine Learning Tools and Techniques" by Ian H. Witten, Eibe Frank, and Mark A. Hall.

Course Code: CSM – 7117 **Credit: 4**
Course Name: Cloud Computing and Internet of Things

Course Objectives:

- To introduce students to the concepts, principles, and models of cloud computing, including infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS), and their applications in various domains.
- To familiarize students with cloud architecture components such as virtualization, containers, orchestration, and microservices, and different deployment models such as public cloud, private cloud, hybrid cloud, and multi-cloud.
- To teach students about various cloud services and technologies, including storage services, compute services, networking services, database services, and serverless computing, and how to leverage them to build scalable and resilient cloud applications.
- To provide students with an understanding of the Internet of Things (IoT) ecosystem, including IoT devices, sensors, actuators, gateways, and communication protocols, and their role in connecting physical objects to the internet.
- To explore various applications and use cases of IoT in different domains such as smart homes, smart cities, healthcare, agriculture, transportation, and industrial automation, and how cloud computing enables IoT data processing, storage, and analysis.

Course Outcome (CO's):

On successful completion of this practical lab session, students will be able to:

1. Proficiency in deploying and managing cloud-based infrastructure and services for scalable and efficient computing.
2. Mastery in developing IoT solutions and integrating sensors, devices, and cloud platforms for data collection and analysis.
3. Skills in implementing security measures and protocols to ensure the confidentiality, integrity, and availability of cloud and IoT systems.
4. Ability to design and optimize cloud and IoT architectures to meet performance, cost, and reliability requirements.
5. Understanding of emerging trends and technologies in cloud computing and IoT for innovation and digital transformation.

Block – I: Cloud Computing Fundamentals and Virtualization

Unit 1: Cloud Computing: Introduction, Traditional Computing Approaches, Evolution of Cloud Computing, Comparison between Cluster, Grid and Cloud Computing, Utility Computing.

Unit 2: Characteristics of Cloud Computing, Benefits of Cloud Computing, Applications of Cloud Computing, Challenges of Cloud Computing.

Unit 3: Cloud Deployment Models, Service Models and Cloud Architecture, Cloud Deployment Models, Public Cloud, Private Cloud, Community Cloud, Hybrid Cloud, Choosing Appropriate Deployment Model, Service Delivery Models of Infrastructure As a Service (IaaS), Platform As a Service (PaaS), Software As a Service.

Unit 4: Resource Virtualization, Virtualization and Underlying Abstraction, Virtualizing Physical Computing Resources, Advantages of Virtualization, Machine or Server Level Virtualization, Hosted

Approach, Bare Metal Approach Exploring Hypervisor or V3irtual Machine Monitor, Hypervisor Based Virtualization Approaches, Operating System Level Virtualization, Other Virtualizations (Network, Storage, Desktop), Xen Server Vs VM ware.

Block – II: Resource Provisioning, Load Balancing and Security

Unit 5: Resource Pooling, Sharing and Provisioning, Resource Pooling, Resource Pooling Architecture, Computer Vs Server Pool, Storage Pool, Network Pool Resource Sharing.

Unit 6: Multi Tenancy, Types of Tenancy, Tenancy at Different Level of Cloud Services, Resource Provisioning and Approaches, Static Approach, Dynamic Approach, Hybrid Approach, VM Sizing,

Unit 7: Scaling, Scaling primitives, Scaling Strategies, Proactive Scaling, Reactive Scaling, Combinational Scaling, Auto Scaling in Cloud, Types of Scaling, Vertical Scaling or Scaling Up, Horizontal Scaling or Scaling Out.

Unit 8: Load Balancing, Importance of Load Balancing, Goals of Load Balancing, What are to Load Balance and how it is done, Levels of Load Balancing, VM Provisioning, Resource Provisioning, Categories of Load Balancing, Static Approach o Dynamic Approach, Dynamic Load Balancing.

Unit 9: Security Issues in Cloud Computing Threats to Cloud Security, Infrastructure Security, Information Security, Identity Management and Access Control, Cloud Security Design Principles, Security as a Service.

Block – III: IoT Fundamentals and Connectivity Technologies

Unit 10: Internet of Things: An Introduction, Introduction to IoT, Characteristics of IoT, IoT Categories, IoT Enablers and Connectivity Layers, Baseline Technologies of IoT, Sensors.

Unit 11: Characteristics of a Sensor, Classification of Sensors, Actuators o Types of Actuators, Computing, IoT Architecture, Applications of IoT, Challenges of IoT.

Unit 12: IoT Networking and Connectivity Technologies, M2M and IoT Technology, Components of Networking, Gateway Prefix Allotment, Impact of Mobility on Addressing.

Unit 13: Multihoming, IoT Identification and Data Protocols, (IPV4, IPv6, MQTT, CoAP, XMPP, AMQP), Connectivity Technologies, (ZigBee, 6LoWPAN, RFID, NFC, Bluetooth, Z-wave etc.)

Block – IV: Application Development, Fog Computing and Case Studies

Unit 14: IoT Application Development, Framework for IoT Applications, Implementation of Device Integration, Data Acquisition and Integration.

Unit 15: Device Data Storage, Unstructured Data Storage on Cloud/Local Server, Authentication, Authorization of Devices, Security Aspects in IoT, Fog Computing and Edge Computing, Introduction to Fog Computing, Cloud Computing Vs Fog Computing, Fog Architecture, Working of Fog, Advantages of Fog, Applications of Fog, Challenges in Fog.

Unit 16: Edge Computing, Working of Edge Computing, Cloud Vs Fog Vs Edge Computing, Applications of Edge Computing. IoT Case Studies • Smart Homes • Smart Grids • Smart Cities • Connected Vehicles • Industrial IoT.

Books Recommended/Suggested Reading:

For Cloud Computing:

1. "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Zaigham Mahmood, and Ricardo Puttini
2. "Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS)

by Michael J. Kavis

For Internet of Things:

3. "Internet of Things: A Hands-On Approach" by Arshdeep Bahga and Vijay Madisetti.

SEMESTER – IV

Course Code: CSM - 7211 **Credit: 3**

Course Name: Big Data Analytics

Course Objective:

- To introduce students to the concept of big data, including its characteristics such as volume, velocity, variety, veracity, and value, and its significance in generating insights and driving decision-making.
- To familiarize students with big data technologies and platforms, including Hadoop, Spark, Kafka, HBase, Cassandra, and MongoDB, and their role in processing, storing, and analysing large volumes of data efficiently.
- To teach students how to collect, clean, pre-process, and transform big data from various sources, including structured and unstructured data, using tools and techniques such as ETL (Extract, Transform, Load) processes, data wrangling, and data integration.
- To enable students to apply data analysis techniques to big data sets, including descriptive analytics, predictive analytics, prescriptive analytics, and advanced analytics such as machine learning and deep learning, to extract valuable insights and patterns.
- To develop students' skills in visualizing and interpreting big data analytics results effectively, using tools and techniques such as data visualization libraries, dashboards, and storytelling, to communicate findings and insights to stakeholders.

Course Outcomes:

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

1. Proficiency in analyzing large-scale datasets to extract valuable insights and patterns.
2. Mastery in utilizing big data tools and technologies for efficient data processing and analysis.
3. Skills in developing predictive models and machine learning algorithms for data-driven decision-making.
4. Ability to interpret and communicate analytical findings to stakeholders effectively.
5. Understanding of ethical considerations and privacy concerns in big data analytics practices.

Block – I: Understanding Big Data and Business Motivations

Unit 1: Introduction to Data Analytics: Sources and nature of data, classification of data (structured, semi-structured, unstructured), characteristics of data, introduction to Big Data platform.

Unit 2: Need of data analytics, evolution of analytic scalability, analytic process and tools, analysis vs reporting, modern data analytic tools, applications of data analytics.

Unit 3: Data Analytics Lifecycle: Need, key roles for successful analytic projects, various phases of data analytics lifecycle – discovery, data preparation, model planning, model building, communicating results, operationalization.

Unit 4: Marketplace dynamics, Business architecture, Business process management, Information and Communications technology.

Block – II: Enterprise Technologies

Unit 5: Data Analysis: Regression modeling, multivariate analysis, Bayesian modeling, inference and Bayesian networks, support vector and kernel methods.

Unit 6: Analysis of time series: linear systems analysis & nonlinear dynamics, rule induction, neural networks: learning and generalization, competitive learning.

Unit 7: Principal component analysis and neural networks, fuzzy logic: extracting fuzzy models from data, fuzzy decision trees, stochastic search methods.

Unit 8: Online Transaction Processing, Online analytical processing, extract transform load, Data warehouses, data marts.

Block – III: Big Data Processing Concepts

Unit 9: Clusters, File Systems and Distributed File systems, NoSQL, Sharding, Replication, Sharding and Replication, CAP Theorem, ACID and BASE.

Unit 10: Mining Data Streams: Introduction to streams concepts, stream data model and architecture, stream computing, sampling data in a stream, filtering streams, counting distinct elements in a stream.

Unit 11: Estimating moments, counting oneness in a window, decaying window, Real-time Analytics Platform (RTAP) applications, Case studies – real time sentiment analysis, stock market predictions.

Block – IV: Big Data Processing Concepts & Techniques

Unit 12: Parallel data processing. Distributed data processing Hadoop, Processing workloads Cluster, Processing in batch mode.

Unit 13: Quantitative analysis, Qualitative analysis. Data Mining

Unit 14: Statistical Analysis, Machine Learning, Semantic Analysis

Books Recommended/Suggested Reading:

1. Big Data: Principles and Best Practices of Scalable Real-Time Data Systems by Nathan Marz and James Warren.

2. Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph" by David Loshin

3. Big Data: A Revolution That Will Transform How We Live, Work, and Think by Viktor Mayer-Schönberger and Kenneth Cukier

4. Big Data: Concepts, Technologies, and Applications by Kuan-Ching Li, Hai Jiang, Laurence T. Yang, and Alfredo Cuzzocrea

5. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data" by EMC Education Services.

Any One Course from the Following 02 (Elective Course – CSM-7212/ CSM-7213)

Course Code: CSM - 7212 Credit: 3
Course Name: Mobile Computing

Course Objective:

- To introduce students to the basic concepts, principles, and characteristics of mobile computing, including mobility, portability, connectivity, and context awareness.
- To teach students how to develop mobile applications for various platforms, including Android and iOS, using programming languages such as Java, Kotlin, Swift, and Objective-C, and mobile development frameworks such as Android SDK, iOS SDK, and React Native.
- To familiarize students with mobile networking technologies and protocols, including cellular networks, Wi-Fi, Bluetooth, NFC (Near Field Communication), and mobile communication standards such as GSM, CDMA, LTE, and 5G.
- To enable students to understand and implement location-based services (LBS) and geospatial technologies in mobile applications, including GPS (Global Positioning System), GIS (Geographic Information System), and location-based advertising.
- To teach students about mobile security threats and vulnerabilities, including malware, phishing, and unauthorized access, and how to implement security measures such as encryption, authentication, and authorization to protect mobile devices and data.

Course Outcomes:

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

1. Proficiency in developing mobile applications for various platforms and devices.
2. Mastery in utilizing mobile computing technologies for location-based services and context-aware applications.
3. Skills in optimizing mobile applications for performance, battery life, and user experience.
4. Ability to design and implement secure and scalable mobile solutions for diverse industries.
5. Understanding of emerging trends and challenges in mobile computing for innovation and digital transformation.

Block – I: Introduction to Mobile Computing

Unit-1: Introduction to Mobile Communications Introduction Objectives Mobile Communication Multiplexing (TDMA, CDMA, FDMA) GSM GPRS and2.5G 3G 4G –LTE.

Unit-2: Introduction to Mobile Computing Architecture Introduction Objectives Mobile IP, Cellular and WLAN IEEE 802.11X Networks.

Unit-3: AdHoc Networks Mobile Computing Operating System Client Server Computing using Mobile Mobile Computing Architecture Design considerations for Mobile Computing Mobile Computing and the Apps Summary Further Readings.

Unit-4: Mobile Client Devices and Pervasive Computing Introduction Objectives Smart Sensors, Actuators and Mobile Robotic Systems Smart Home and Appliances Automotive Systems Limitations and Devices Design Considerations.

Unit-5: GSM and GPRS Introduction Objectives GSM Architecture Public Land Mobile Network (PLMN) Interface Call Handling Handover SMS GPRS High Speed Circuit Switched Data WLL Application Summary Further Readings

Block – II: Mobile IP and Issues in Mobile Computing

Unit-6: 4G and 5G Networks Introduction Objectives High Speed Packet Access MIMO in HSPA LTE and WIMAX16E Ultra-Wide Band and Broadband Wireless Access 4G Networks: HS-OFDM, LTE Advanced and WiMax 16M Features of 5G Networks.

Unit-7: Mobile IP Network Layer Introduction Objectives Mobile IP IP Header: Encapsulation and Routes Optimization Mobility Binding Cellular IP Mobile IP with IPv6 Voice over IP, IP Security.

Unit-8: Mobile Transport Layer Introduction Objectives, UDP and TCP Indirect TCP Snooping TCP Mobile TCP.

Unit-9: Database Management Issues in Mobile Computing Introduction Mobile Device Database Management Mobile Device Data Store Methods Client Server Computing with Adaptation for Mobile Computing Adaptation Software for Mobile Computing.

Block – III: Introduction to various Network Technologies

Unit-10: Mobile Adhoc Networks Introduction Objectives Introduction to MANETs Routing and Classifications of Routing Algorithms QoS in MANETs Security in MANETs.

Unit-11: WLAN and PAN protocols Introduction Objectives Introduction to WLANs Introduction to WAP Introduction to WML Bluetooth Wi-Max ZigBee and Wi-Fi.

Unit-12: Virtual and Cloud Networks Introduction Objectives Wireless Enterprise Networks Virtual Networks Mobile Cloud Networks.

Unit-13: Mobility, Portability, Replication and Clustering Introduction Objectives Mobile Data Management Data Replication Schemes Adaptive Clustering.

Block – IV: Introduction to Mobile Software Environments

Unit-14: Smart Client and Enterprise Server based Architecture Introduction Objectives Introduction to Smart Client Architecture Data Synchronization Formats Data Synchronization at Clients and Servers Mobile Devices Support Infrastructure and Management.

Unit-15: Mobile Internet Applications Introduction Objectives Introduction to Mobile Applications Development Introduction to XML Handheld Device Markup Language and WML HTML 5

Unit-16: Mobile Application Languages Introduction, Mobile Operating Systems and Development Environments Introduction Objectives Introduction to Mobile Operating Systems Application Programming, Linux for Mobile Devices Development Process Development Tools.

Books Recommended/Suggested Reading:

1. Mobile Computing: Principles and Practices" by Reza B'Far.
2. Mobile Computing" by Asoke K. Talukder and Roopa R. Yavagal.
3. Android Programming: The Big Nerd Ranch Guide" by Bill Phillips, Chris Stewart, Kristin Marsicano.
4. Fundamentals of Mobile and Pervasive Computing" by Frank Adelstein, Sandeep K.S. Gupta, Golden G. Richard III, and Loren Schwiebert.
5. Mobile Computing: Technology, Applications, and Service Creation" by Amjad Umar.

Course Code: CSM - 7213 **Credit: 3**

Course Name: Deep Learning

Course Objective:

- To provide students with a foundational understanding of deep learning concepts, architectures, and algorithms, including artificial neural networks (ANNs), deep neural networks (DNNs), and deep learning frameworks.
- To teach students about various deep learning models and architectures, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory (LSTM) networks, and generative adversarial networks (GANs), and their applications in computer vision, natural language processing, and other domains.
- To familiarize students with popular deep learning frameworks and tools, such as TensorFlow, Keras, PyTorch, and Caffe, and how to use them to build, train, and deploy deep learning models efficiently.
- To enable students to understand and apply training and optimization techniques for deep learning models, including stochastic gradient descent (SGD), mini-batch gradient descent, learning rate scheduling, regularization, and dropout.
- To explore various applications and use cases of deep learning in different domains, including image recognition, object detection, speech recognition, machine translation, recommendation systems, and autonomous driving, and how deep learning is revolutionizing industries and transforming society.

Course Outcomes:

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

1. Proficiency in implementing advanced neural network architectures for complex problem-solving.
2. Mastery in applying deep learning techniques to tasks such as image recognition and natural language processing.
3. Skills in training and fine-tuning deep learning models for improved performance and accuracy.
4. Ability to analyze and interpret deep learning results to derive actionable insights.
5. Understanding of emerging trends and applications in deep learning for innovation and advancement in AI.

Block – I: Introduction to Deep Learning and Neural Network

Unit-1: Fundamentals of Deep Learning: Overview of deep learning and its evolution from traditional machine learning, highlighting key differences and advancements. Introduction to artificial neural networks (ANNs) and the significance of deep networks in handling complex data, Key concepts, Applications of deep learning in various fields.

Unit-2: Basics of Neural Networks: Structure and working of single-layer and multi-layer neural networks, Forward propagation, backpropagation, and the process of training neural networks using gradient descent, Types of neural networks: feedforward neural networks, convolutional neural networks (CNNs), and recurrent neural networks (RNNs).

Unit-3: Activation Functions and Optimization Techniques: Importance of activation functions in introducing non-linearity to neural networks, Common activation functions, Overview of optimization

algorithms such as Gradient Descent, Stochastic Gradient Descent, and Adam Optimizer, Techniques to improve learning and convergence and dropout.
Unit-4: Introduction to Deep Learning Frameworks: Hands-on introduction to popular deep learning frameworks like TensorFlow, Keras, and PyTorch, Basic operations in these frameworks: setting up a model, defining layers, compiling, and training.

Block – II: Convolutional Neural Network (CNNs)
Unit-5: Fundamentals of Convolutional Neural Networks (CNNs): Introduction to CNN architecture, including concepts of convolutional layers, filters, and feature maps. Explanation of CNNs differ and Well-suited for image processing. Overview of concepts such as receptive fields, kernel size.
Unit-6: Pooling and Regularization Techniques in CNNs: Understanding pooling layers, including max pooling and average pooling, to reduce spatial dimensions and control over fitting. Explanation of regularization techniques specific to CNNs.
Unit-7: Popular CNN Architectures and Applications: Exploration of widely used CNN architectures, such as LeNet, AlexNet, VGG, ResNet, and Inception. Applications of CNNs in real-world scenarios, including image classification, object detection, and facial recognition.
Unit-8: Transfer Learning and Fine-Tuning in CNNs: Introduction to transfer learning, Pre-trained CNN models are adapted to new tasks with limited data. Explanation of fine-tuning techniques and Hands-on implementation using popular frameworks to apply transfer learning with CNNs on customized datasets.

Block – III: Recurrent Neural Network (RNNs) and Sequence Modeling
Unit 9: Introduction to Sequence Modeling and Recurrent Neural Networks (RNNs): Overview of sequence data and time-series applications in natural language processing, speech recognition, and predictive analytics. Fundamentals of RNNs: structure, function, and significance in processing sequential data.
Unit 10: Advanced RNN Architectures: Challenges of traditional RNNs: vanishing and exploding gradients. Introduction to advanced RNN architectures: Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs).
Unit 11: Sequence-to-Sequence Models and Attention Mechanisms: Sequence-to-Sequence modeling for tasks requiring input-output sequence mapping, such as translation and summarization. Encoder-Decoder architecture and its role in Seq2Seq models.
Unit 12: Applications and Practical Implementation of RNNs in Deep Learning Frameworks: Hands-on implementation of RNN, LSTM, and GRU models using deep learning frameworks such as TensorFlow and PyTorch. Practical applications in time-series prediction, financial forecasting, and sequential data analysis.

Block – IV: Advanced Deep Learning Techniques and Optimization
Unit-13: Advanced Neural Network Architectures: Exploration of advanced architectures beyond basic neural networks, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Autoencoders. Introduction to Generative Adversarial Networks (GANs): architecture, training, and applications in image synthesis and data augmentation.

Unit-14: Optimization Techniques in Deep Learning: Fundamentals of optimization in neural networks, focusing on gradient descent and its variants. Advanced optimizers: RMSprop, Adam, and Adagrad, discussing their role in accelerating convergence and improving model accuracy.
Unit-15: Transfer Learning and Pretrained Models: Introduction to transfer learning and its significance in improving efficiency, particularly in scenarios with limited data. Fine-tuning of pretrained models and exploration of popular models like VGG, ResNet, BERT, and GPT.
Unit-16: Advanced Topics: Reinforcement Learning and Deep Reinforcement Learning: Basics of Reinforcement Learning (RL): concepts of agents, environments, rewards, and policy learning. Introduction to Deep Reinforcement Learning (DRL) and the combination of deep neural networks with RL algorithms.

Books Recommended/Suggested Reading:

1. Bishop C.M-Pattern Recognition and Machine Learning-Springer, 2006.
2. Yegnanarayana. B- Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.
3. Golub, G., H., and Van Loan, C., F., Matrix Computations-JHU Press,2013.
4. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004.

Course Code: CSM - 7214 **Credit: 3**
Course Name: Natural Language Processing

Course Objective:

- To understand the fundamental concepts and techniques of Natural Language Processing.
- To explore various text processing methods, including tokenization, stemming, and lemmatization.
- To implement NLP tasks such as Named Entity Recognition, Part-of-Speech Tagging, and Parsing.
- To apply machine learning and deep learning models for text classification and sentiment analysis.
- To develop applications using NLP libraries like NLTK, Spacy, and Transformers.

Course Outcomes:

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

1. Demonstrate proficiency in text preprocessing techniques for NLP applications.
2. Implement various NLP models and algorithms for language understanding.
3. Analyze and extract meaningful insights from textual data using statistical and ML approaches.
4. Develop real-world applications such as chatbots, sentiment analyzers, and translation systems.
5. Utilize advanced NLP frameworks and deep learning techniques for complex language processing tasks.

Block-I: Introduction to Natural Language Processing

Unit 1: Fundamentals of NLP – Introduction, History, Applications, and Challenges.

Unit 2: NLP Pipeline – Tokenization, Stemming, Lemmatization, Stopword Removal.

Unit 3: Language Modeling – N-grams, Smoothing Techniques, Perplexity, Word Embeddings.

Unit 4: Text Preprocessing Techniques – POS Tagging, Named Entity Recognition (NER), Chunking.

Block-II: Syntax, Semantics, and Parsing Techniques

Unit 5: Syntactic Analysis – Context-Free Grammar (CFG), Dependency Parsing.

Unit 6: Statistical Parsing – CYK Algorithm, Earley's Algorithm.

Unit 7: Semantic Analysis – Lexical Semantics, Word Sense Disambiguation (WSD), Ontologies.

Unit 8: Vector Space Models – TF-IDF, Word2Vec, GloVe, FastText.

Block-III: Advanced NLP Techniques

Unit 9: Sentiment Analysis and Text Classification – Naïve Bayes, SVM, LSTMs, BERT.

Unit 10: Named Entity Recognition (NER) and Relation Extraction.

Unit 11: Topic Modeling – Latent Dirichlet Allocation (LDA), Non-Negative Matrix Factorization (NMF).
Unit 12: Machine Translation – Statistical and Neural Machine Translation (NMT), Transformers.

Block-IV: NLP Applications and Deep Learning in NLP
Unit 13: Speech Recognition and Text-to-Speech (TTS) Systems.
Unit 14: Question Answering Systems and Chatbots – Retrieval-based and Generative Models.
Unit 15: Summarization – Extractive and Abstractive Approaches.
Unit 16: Ethical Considerations and Bias in NLP – Fairness, Privacy, and Responsible AI.

Books Recommended/Suggested Reading:

1. Speech and Language Processing – Daniel Jurafsky and James H. Martin: A comprehensive introduction to NLP, covering linguistic, statistical, and deep learning approaches.
2. Foundations of Statistical Natural Language Processing – Christopher D. Manning and Hinrich Schütze, A fundamental book on statistical methods for NLP, including text classification and language modeling.
3. Neural Network Methods for Natural Language Processing – Yoav Goldberg, Explores deep learning techniques in NLP, including embeddings, recurrent networks, and attention mechanisms.
4. Natural Language Processing with Python – Steven Bird, Ewan Klein, and Edward Loper, A practical guide using Python and NLTK for text processing, parsing, and classification.
5. Deep Learning for Natural Language Processing – Palash Goyal, Sumit Pandey, and Karan Jain Covers deep learning-based NLP models, including LSTMs, Transformers, and BERT, with practical implementations.

Course Code: CSM - 7215 **Credit: 2**
Course Name: Python Programming

Course Objective:

- To introduce the fundamental concepts and syntax of Python programming.
- To develop problem-solving skills using Python for real-world applications.
- To implement object-oriented programming concepts in Python.
- To utilize Python libraries for data processing, visualization, and automation.
- To enhance coding efficiency and debugging skills for software development.

Course Outcomes:

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

1. Proficiency in Python programming language syntax, data structures, and algorithms.
2. Mastery in developing Python applications for various domains and industries.
3. Skills in debugging, testing, and optimizing Python code for efficiency and reliability.
4. Ability to apply Python programming concepts to solve real-world problems and tasks.
5. Understanding of best practices and software development principles in Python programming.

Block – I: Python Basics, Conditional, Loops, String Objects and List Objects

Unit 1: Introduction to Python: History of Python, Need of Python, Packages for Cross platform application of Python, Getting started with Python, Program structure in python, Running the First program.

Unit 2: Installation of Python and python Notebook, Python Objects, Number & Booleans, Strings, Container objects, Mutability of objects, Operators - Arithmetic, Bitwise.

Unit 3: comparison and Assignment operators, Operators Precedence and associativity. Conditions (If else, if-elif, else), Loops (While, for), Break and Continue statements, Range Functions.

Unit 4: String object basics, String methods, Splitting and Joining Strings, String format functions, list object basics, list methods, List as stack and Queues, List comprehensions.

Block – II: Tuples, Set, Dictionaries & Functions, Modules, Exception Handling

Unit 5: Functions and File Handling in Python: Function definition and call, Function Scope, Arguments, Function Objects, Lambda Functions, Anonymous Functions, File Operations, Creating. Opening and using files.

Unit 6: Tuples, Sets, Dictionary Object basics, Dictionary Object methods, Dictionary View Objects.

Unit 7: Functions basics, Parameter passing, Iterators, Generator functions, Lambda functions, Map, Reduce, filter functions.
Unit 8: OOPS Concepts & Working with Files OOPS: Basic concepts, creating classes and Objects. Inheritance, Multiple Inheritance, working with files, Reading and writing files, Buffered read and write, Other File methods.
Unit 9: Modules and Packages: Module Creations and Usage, Module Search Path, Module Vs. Script, Package Creation and Importing, Standard Library Modules.
Unit 10: Using Standard Module, Creating new modules, Exceptions Handling with Try-except, Creating, inserting and retrieving Table, Updating and deleting the data.
Unit 11: Data Ananlysis- Numpy variable, Numpy manipulation, Scipy, Pandas intro. Descriptive analysis, Pandas Input-output, Pandas manipulation, Pandas groupby.
Unit 12: Modules and Packages: Module Creations and Usage, Module Search Path, Module Vs. Script, Package Creation and Importing, Standard Library Modules.

Books Recommended/Suggested Reading:

1. Head First Python 2e: A Brain-Friendly Guide Paperback – Illustrated, 16 by Paul Barry, Oreilly.
2. Python: The Complete Reference Paperback – 20 March 2018 by Martin C. Brown (Author), TMH Publication
3. Let Us Python by Yashavant Kanetkar , 1 January 2019, BPB publication.
4. Python Programming, A modular approach, First Edition, By Pearson Publication by Taneja Sheetal and Kumar Naveen, 26 September 2017.

Course Code: CSM - 7216 **Credit: 3**
Course Name: Quantum Computing

Course Objective:

- To understand the fundamental principles of quantum computing and its distinction from classical computing.
- To explore quantum mechanics concepts such as superposition, entanglement, and quantum gates.
- To analyze and implement quantum algorithms like Shor's and Grover's for real-world applications.
- To study quantum error correction techniques and hardware implementations.
- To examine the future scope of quantum computing in cryptography, AI, and data science.

Course Outcomes:

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

1. Demonstrate a deep understanding of quantum computing principles and their applications.
2. Apply quantum algorithms to solve complex computational problems efficiently.
3. Implement quantum programs using frameworks like Qiskit and Cirq.
4. Evaluate the challenges of quantum error correction and hardware limitations.
5. Investigate emerging trends in quantum cryptography, communication, and AI.

Block-I: Introduction to Quantum Computing

Unit 1: Basics of Quantum Computing – History, Motivation, and Applications
Unit 2: Classical vs. Quantum Computing – Key Differences and Advantages
Unit 3: Qubits and Quantum States – Superposition, Entanglement, and Measurement
Unit 4: Quantum Gates and Circuits – Quantum Logic Gates (Pauli, Hadamard, CNOT, Toffoli, etc.)

Block-II: Quantum Algorithms and Computation Models

Unit 5: Quantum Parallelism and Quantum Speedup
Unit 6: Quantum Fourier Transform and its Applications
Unit 7: Shor's Algorithm – Integer Factorization and Cryptography Applications
Unit 8: Grover's Algorithm – Quantum Search Algorithm and Optimization Problems

Block-III: Quantum Error Correction and Hardware Implementations

Unit 9: Quantum Decoherence and Noise – Challenges in Quantum Computing
Unit 10: Quantum Error Correction – Concepts, Techniques, and Codes
Unit 11: Physical Realization of Quantum Computers – Superconducting Qubits, Ion Traps,

Topological Qubits.
Unit 12: Quantum Computing Frameworks – Qiskit, Cirq, and Microsoft Q#

Block-IV: Advanced Topics and Future Directions in Quantum Computing
Unit 13: Quantum Cryptography – Quantum Key Distribution (QKD), BB84 Protocol
Unit 14: Quantum Machine Learning – Applications in AI and Data Science
Unit 15: Quantum Internet and Communication – Concepts and Emerging Trends
Unit 16: Future of Quantum Computing – Challenges, Opportunities, and Ethical Considerations

Books Recommended/Suggested Reading:

1. "Quantum Computation and Quantum Information" – Michael A. Nielsen and Isaac L. Chuang.
2. "Quantum Computing for Computer Scientists" – Noson S. Yanofsky and Mirco A. Mannucci.
3. "An Introduction to Quantum Computing" – Phillip Kaye, Raymond Laflamme, and Michele Mosca.
4. "Quantum Computing: A Gentle Introduction" – Eleanor G. Rieffel and Wolfgang H. Polak.
5. "Quantum Computing Explained" – David McMahon.

Course Code: CSM - 7251 **Credit: 2**
Course Name: Python Programming Lab

Course Objective:

- To develop proficiency in writing Python programs for problem-solving.
- To implement control structures, functions, and modules in Python.
- To apply object-oriented programming concepts using Python.
- To work with file handling, data structures, and exception handling.
- To integrate Python libraries for data manipulation and visualization.

Course Outcomes:

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

1. Demonstrate the ability to write, debug, and execute Python programs efficiently.
2. Implement algorithms using Python's built-in data structures and libraries.
3. Develop modular and reusable code using functions and object-oriented principles.
4. Apply file handling and exception handling techniques in real-world applications.
5. Utilize Python for data processing, analysis, and visualization tasks.

Python Programming Lab:

Basic Programs

1. **Hello World Program** – Print "Hello, World!" to the console.
2. **Simple Calculator** – Perform basic arithmetic operations (addition, subtraction, multiplication, division).
3. **Even or Odd Checker** – Check whether a given number is even or odd.
4. **Factorial Calculation** – Compute the factorial of a given number using recursion.
5. **Prime Number Checker** – Determine whether a given number is prime.
6. **Fibonacci Series** – Generate the Fibonacci sequence up to a specified number.
7. **Reverse a String** – Reverse a given string without using built-in functions.
8. **Palindrome Checker** – Check whether a given string or number is a palindrome.
9. **Simple List Operations** – Perform insert, delete, sort, and search operations on a list.
10. **Student Grade Calculator** – Take marks of different subjects and calculate total, percentage, and grade.

Advanced Programs

11. **File Handling Operations** – Read and write data to a text file.
12. **Bank Account Management System** – Simulate deposit, withdrawal, and balance inquiry using OOP concepts.
13. **Matrix Operations** – Perform addition, subtraction, multiplication, and transpose of matrices.

14. **Regular Expressions** – Extract emails, phone numbers, and special patterns from a text using regex.
15. **Web Scraping** – Extract data from a website using the BeautifulSoup library.
16. **Data Visualization** – Generate bar charts and pie charts using Matplotlib.
17. **Multithreading Example** – Demonstrate thread creation and synchronization.
18. **Chatbot Simulation** – Create a simple chatbot using predefined responses.
19. **API Integration** – Fetch data from an external API using the requests library.
20. **Machine Learning Model (Basic)** – Implement a simple linear regression model using Scikit-learn.

Recommended/Suggested Reading

1. **"Python Crash Course"** by Eric Matthes – A hands-on, project-based introduction to Python programming.
2. **"Learning Python"** by Mark Lutz – A comprehensive guide covering Python fundamentals and advanced topics.
3. **"Python Programming: An Introduction to Computer Science"** by John Zelle – A foundational book for learning Python in a computer science context.
4. **"Automate the Boring Stuff with Python"** by Al Sweigart – A practical guide for automating tasks using Python.
5. **"Think Python: How to Think Like a Computer Scientist"** by Allen B. Downey – A structured approach to learning Python with problem-solving techniques.

Course Code: CSM - 7291 **Credit: 4**
Course Name: Project / Dissertation

Course Objective:

- The primary objective of the project/dissertation is to provide students with an opportunity to apply the knowledge and skills acquired throughout the MCA program to solve real-world problems or explore new research areas within the field of computer applications.
- To develop students' research skills, including literature review, problem identification, hypothesis formulation, research design, data collection, analysis, interpretation, and drawing conclusions, through hands-on experience in conducting independent research.
- To enhance students' project management skills, including project planning, scheduling, resource allocation, risk management, and communication, enabling them to effectively manage the entire project lifecycle from initiation to completion.
- To cultivate students' problem-solving and critical thinking abilities by challenging them to identify, analyze, and address complex problems or research questions, using innovative and creative approaches.
- To improve students' communication and presentation skills, both written and oral, by requiring them to document their research findings, methodology, and outcomes effectively, and present them to a diverse audience in a clear, concise, and engaging manner.

Course Outcomes:

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

1. Demonstration of proficiency in applying theoretical knowledge and practical skills to solve real-world problems or conduct original research.
2. Mastery in analyzing, designing, implementing, and evaluating software solutions or research methodologies.
3. Development of scholarly and systematic approaches to project management and execution.
4. Contribution to the advancement of knowledge and innovation in the field of computer applications.
5. Enhancement of professional skills and experience for future career opportunities in academia, industry, or research.

MOOCs

The University shall give flexibility in opting for MOOC (Massive Online Open Courses)/SWAYAM by the students pertaining to the prescribed curriculum and also the credits earned in the MOOC courses may be dealt as part of the evaluation scheme as per UGC (Open and Distance Learning Programmes and Online Programmes) Regulations, 2020.

Syllabi and Course Materials

The Syllabi, PPR and Self-Learning Materials (SLMs) are developed mostly by experienced faculty members of **Mata Tripura Sundari Open University** in consultation with content experts and the same will be forwarded to CIQA and BOS/Academic Council/ Executive Council for further suggestions and approval.

Faculty and Support Staff

The University has identified the dedicated requisite faculty and support staff as mandated by the UGC and they are allocated the positions exclusively for ODL mode. The course material prepared by the CDOE faculty is at par with the regulations 2020.

List of Faculty associated with MCA programme is as follows:

S.No.	Name of Faculty	Designation	Nature of Appointment	Qualification	Subject
1.	Dr. Aasheesh Raizada	Associate Professor	Full-Time	Ph.D.	Computer Science
2.	Mr. Pankaj Agarwal	Assistant Professor	Full-Time	UGC NET	Computer Science

Delivery Mechanism

The ODL of MTSOU follows a modern ICT (Information & Communication Technology) enabled approach for instruction. The methodology of instruction in ODL of MTSOU is different from that of the conventional/regular programs. Our ODL system is more learner-oriented and the learner is an active participant in the teaching-learning process. ODL of MTSOU academic delivery system comprises:

A. Print Material

The printed material of the programme supplied to the students will be unit wise for every course.

B. Counselling Sessions

Normally, counselling sessions are held as per a schedule drawn beforehand by the Subject Coordinator. There will be 6 counselling/ contact classes for 4 credit course will be held on the campus on Saturday and on Sunday of 2 hour duration for each course in face to face mode (In case of 2 credit course contact hours are required 6 hours and in case of 6 credit course contact hours required 18 hours). Contact classes will be held in the campus on Saturdays and on Sundays.

C. Medium of Instruction

Medium of Course Instruction: English

Medium of Examination: English

D. Student Support Systems

Universities Study Centres or Learner Support Centre shall be headed by a coordinator, not below the rank of Assistant professor and shall be augmented with academic and non-academic staff depending on the learner.

The university has made appropriate arrangements for various support services including counselling schedule and resource-oriented services evaluation methods and dates both online and offline modes for easy and smooth services to the students of distance mode.

At present the university have only one study centre on the campus. The institution is not promoting any study centres outside the campus. All student support services will be provided to the student through a single window method/mode onsite and online.

E. Procedure for Admissions, Curriculum, Transaction and Evaluation

Admission Process

Admission to the MCA Programme will be done on the basis of screening of candidate's eligibility on first come first serve basis. The University will follow the reservation policy as per norms of the Government. Admission shall not be a right to the students and MTSOU, DDOE

shall retain the right to cancel any admission at any point of time if any irregularity is found in the admission process, eligibility etc.

Maximum Duration

A. The maximum duration of the MCA Programme is four years. Thereafter, students seeking completion of the left-over course(s) will be required to seek fresh admission.
B. The student can complete his programme within a period of 4 years failing which he/she shall seek fresh admission to complete the programme.

Eligibility

BCA or equivalent course from recognized board.

Fee Structure

Name of the Program	Degree	Duration	Year	Tuition Fee / Year	Exam Fee /Year	Total (in Rs.)
Master of Computer Application	PG	2 to 4 Years	1	30000	1500	31500
			2	30000	1500	31500
TOTAL				**60000**	**3000**	**63000**

Activity Schedule

S.NO.	Name of the Activity	Tentative months schedule (specify months) during year			
		From(Month)	To (Month)	From(Month)	To (Month)
1	Admission	Jul	Sep	Jan	Mar
2	Assignment submission (if any)	Sep	Oct	Mar	Apr
3	Evaluation of assignment	Oct	Nov	Apr	May
4	Examination	Dec	Dec	Jun	Jun
5	Declaration of result	Jan	Jan	Jul	Jul
6	Re-registration	Jul	Jul	Jan	Jan
7	Distribution of SLM	Jul	Sep	Jan	Mar
8	Contact programmes (counselling, practicals, etc.)	Sep	Nov	Mar	May

Credit System

MTSOU, CDOE proposes to follow the 'Credit System' for most of its programs. Each credit amounts to 30 hours of study comprising all learning activities. Thus, a 8 credit course requires 240 hours, 6 credit course requires 180 hours , 4 credit course requires 120 hours and 2 credit course requires 60 hours of study. This helps the student to understand the academic effort to complete a course. Completion of an academic programme requires successful clearing of both, the assignments and the term-end examination of each course in a programme.

Duration of the Programme	**Credits**	**Name of the Programme**	**Level of the Programme**
2 to 4 Yrs	80	MCA	Master's Degree

Assignments

Distance Education learners have to depend much on self study. In order to ascertain the writing skill and level of comprehension of the learner, assignment work is compulsory for all learners. Each assignment shall consist of a number of questions, case studies and practical related tasks. The assignment question papers will be uploaded to the website within a scheduled time and the learners shall be required to respond them within a specified period of time. The response of the learner is examined by a faculty member.

Evaluation

The evaluation system of the programme is based on two components:

A. Continuous evaluation in the form of assignments (weightage 30%): This Component carries a weightage of 30%. There will be at least one graded assignment and test per course. These assignments are to be submitted to the Co-ordinator of the DDOE/Study Centre to which the student is assigned or attached with.

B. Term-end examination (weightage 70%): This will be held twice every year in the months of June and December. The students are at liberty to appear in any of the examinations conducted by the University during the year. A student will be allowed to appear in the Term-End Examination only after she/he has registered for that course and submitted the assignment. For appearing in the Examination, every student has to submit an Examination form through online or offline before the due dates as given in the schedule of operations. If a student misses any term-end examination of a course for any reason, s/he may appear for any of them or all the courses subject to the maximum of 8 courses in the subsequent term-end examinations. This facility will be available until a student secures the minimum pass grade in the courses but up to a maximum period of four semesters, since the date of registration of the course is valid for four semesters. Beyond this period s/he may continue for another four semesters by getting Re-registration by paying fee again. In that case, the score of qualified assignments and/or term-end examination will be retained and the student will be required to complete the left out requirements of such re-registered courses. Minimum requirement for passing a course will be 40% marks.

C. Laboratory Support and Library Resources

The library of Mata Tripura Sundari Open University aims to empower the teaching mission and intellectual culture of the community through availability through an organized collection of information as well as instruction in its access, relevance and evaluation. The University Library enriches advance learning and discovery by providing access to a broad array of resources for education, research and creative work to ensure the rich interchange of ideas in the pursuit of knowledge.

The Directorate of Distance Education of Mata Tripura Sundari Open University has initiated the process of setting up a dedicated Library for ODL program and acquiring printed books and e-books for this purpose. The required International and National subject journals are also provided. We have a full functioning community radio service onboard (FM). We already have annual journal subscriptions and the capacity can be enlarged at later stages as the University lines up with more online journals.

The collection of the Library is rich and diverse especially in terms of the breadth and depth of coverage. Collection encompasses subjects in Management, Commerce, Information Technology, Computer Applications, and other allied areas. This collection further includes Books, Research Journals, Project Reports/Dissertations and online Journals.

The University has well equipped Computer Laboratories, Lecture Capturing Systems, Audio Video facilities, ICT enabled class rooms, Wi-Fi facilities etc.

D. Cost Estimate of the programme and the provisions

Initial expenses have been done by the University in terms of provision of infrastructure, manpower, printing of Self Study Material etc. The University intends to allocate expenses out of the total fee collection as per following details:

a) SLM Development and Distribution	:	20%
b) Postal and ICT Expenses	:	10%
c) Salary and other Administrative expenses	:	60%
d) Future Research development reserve	:	10%
e) Lab Instruments	:	10%

Once programmes are operational, the programme budget from fee receipts will be planned as per the guidelines of University Grants Commission.

I. Quality Assurance

The University has established the Centre for Internal Quality Assurance (CIQA) in the University campus. The CIQA will monitor and maintain the quality of the ODL programmes. It has the following objectives in making the compliances of quality implementations.

Objectives

The objective of Centre for Internal Quality Assurance is to develop and put in place a comprehensive and dynamic internal quality assurance system to ensure that programmes of higher education in the Open and Distance Learning mode and Online mode being implemented by the Higher Educational Institution are of acceptable quality and further improved on continuous basis.

Functions of CIQA

The functions of Centre for Internal Quality Assurance would be following:

1) To maintain quality in the services provided to the learners.
2) To undertake self-evaluative and reflective exercises for continual quality improvement in all the systems and processes of the Higher Educational Institution.
3) To contribute in the identification of the key areas in which Higher Educational Institution should maintain quality.
4) To devise mechanism to ensure that the quality of Open and Distance Learning programmes and Online programmes matches with the quality of relevant programmes in conventional mode.
5) To devise mechanisms for interaction with and obtaining feedback from all stakeholders namely, learners, teachers, staff, parents, society, employers, and Government for quality improvement.
6) To suggest measures to the authorities of Higher Educational Institution for qualitative improvement.
7) To facilitate the implementation of its recommendations through periodic reviews.
8) To organize workshops/seminars/symposium on quality related themes, ensure participation of all stakeholders, and disseminate the reports of such activities among all the stakeholders in Higher Educational Institution.

9) To develop and collate best practices in all areas leading to quality enhancement in services to the learners and disseminate the same all concerned in Higher Educational Institution.
10) To collect, collate and disseminate accurate, complete and reliable statistics about the quality of the programme(s).
11) To ensure that Programme Project Report for each programme is according to the norms and guidelines prescribed by the Commission and wherever necessary by the appropriate regulatory authority having control over the programme;
12) To put in place a mechanism to ensure the proper implementation of Programme Project Reports.
13) To maintain a record of Annual Plans and Annual Reports of Higher Educational Institution, review them periodically and generate actionable reports.
14) To provide inputs to the Higher Educational Institution for restructuring of programmes in order to make them relevant to the job market.
15) To facilitate system based research on ways of creating learner centric environment and to bring about qualitative change in the entire system.
16) To act as a nodal coordinating unit for seeking assessment and accreditation from a designated body for accreditation such as NAAC etc.
17) To adopt measures to ensure internalization and institutionalization of quality enhancement practices through periodic accreditation and audit.
18) To coordinate between Higher Educational Institution and the Commission for various qualities related initiatives or guidelines.
19) To obtain information from other Higher Educational Institutions on various quality benchmarks or parameters and best practices.
20) To record activities undertaken on quality assurance in the form of an annual report of Centre for Internal Quality Assurance.
21) It will be mandatory for Centre for Internal Quality Assurance to submit Annual Reports to the Statutory Authorities or Bodies of the Higher Educational Institution about its activities at the end of each academic session. A copy of report in the format specified by the Commission, duly approved by the statutory authorities of the Higher Educational Institution shall be submitted annually to the Commission.

After enrolling in MCA Programme at Mata Tripura Sundari Open University in ODL mode, student will exhibit understanding in areas such as critical thinking, effective communication and develop problem solving, scientific temperament with right set of ethics and attitude towards environment and sustainability. After completion of MCA Programme, student will participate in multiple functional areas of science and technology.