Master of Computer Application
Open and Distance Learning
(MCA-ODL)

PROGRAMME PROJECT REPORT
(PPR)

Chhatrapati Shahu Ji Maharaj University (CSJMU)
UP State University | Formerly Kanpur University
Accredited 'A++' by NAAC | UGC Approved Category 1 University
Kanpur (UP)

About the Programme

The Master of Computer Application (MCA) – Open and Distance Learning (ODL) programme offered by Chhatrapati Shahu Ji Maharaj University, Kanpur allowing students to study remotely without the need to attend traditional in-person classes. These programs are often designed to accommodate the needs of working professionals or individuals who are unable to commit to a full-time, on-campus program due to various reasons such as job commitments, family responsibilities, or geographical constraints. CSJM University, a Category-1 and accredited as 'A++' by NAAC, university is offering those students a best and easy path to develop their skills. The university has experienced faculty members, an excellent library, and other modern facilities to provide a proper learning environment for the students. This programme is very well received by industry. This is a 02 Year (04-Semester) programme. This programme is designed in such a way to equip students with a holistic set of skills and competencies essential for success in the field of information technology and focuses on imparting to students the ability to demonstrate leadership, understand human relationships, and problem-solving abilities essential for success in IT/ Corporate world.

Vision of the University

To enlighten and empower humanity by nurturing future leaders and change agents for universal development and societal transformation.

Mission of the University

To work towards sustainable excellence in global standards of academia, technology-centric learning, robust research ecosystem, institutional distinctiveness, and harmonious social diversity.

I. Mission and Objective of Master of Computer Application (MCA) Programme:

The mission and objectives of MCA-ODL Programme would be tailored to cater to a diverse range of learners who seek accessible, flexible, and high-quality education in computer application. Here's a proposed framework for the mission and objectives:

1. **Mission:**

   To provide a comprehensive and innovative MCA Programme aims to prepare students for success in the information technology industries all over world by equipping them with relevant knowledge, skills, and competencies. The mission is to foster not only academic growth but also personal and professional development. This may include opportunities for internships, industry partnerships, and career services support.

2. **Objectives:**

   - ***Accessibility:*** To offer high-quality education in computer applications to individuals who face obstacles attending traditional on-campus programs due to geographical constraints, work commitments, or personal circumstances.
   - ***Flexibility:*** To offer flexible scheduling options that accommodate the diverse needs of distance learners, allowing them to balance their studies with work, family, and other responsibilities.
   - ***Engagement:*** To foster active engagement and collaboration among students, instructors, and course content through the effective use of online learning technologies, discussion forums, virtual classrooms, and interactive multimedia resources.
   - ***Skill Development:*** This programme aims to enhance students' analytical, critical thinking, problem-solving, communication, and teamwork skills, ensuring they are well-equipped to excel in the dynamic field of computer applications.
   - ***Technological Proficiency:*** To equip students with advanced skills in utilizing digital tools and technologies essential for various business and industry applications. This

includes proficiency in utilizing online learning platforms, mastering data analysis software, and effectively leveraging communication tools to thrive in the rapidly evolving landscape of information technology operations.

- ***Global Perspective:*** To expose MCA students to a diverse range of global perspectives in the field of computer applications, including exploring emerging technologies, international IT markets, and cultural nuances. This includes understanding the impact of globalization on technology-driven businesses, adapting to cross cultural communication and collaboration, and navigating the complexities of global IT ecosystems.
- ***Carrier Readiness:*** To equip MCA students with the necessary skills and knowledge for entry-level positions in diverse fields of the IT industry or to pursue further education at the graduate level. This is achieved through the provision of comprehensive career development resources, opportunities for internships, and avenues for networking with industry professionals.
- ***Continuous Improvement:*** To continuously evaluate and improve the program based on feedback from students, instructors, employers, and industry trends, ensuring that it remains relevant and effective in meeting the needs of learners and the demands of the business and industry environment.

Program Outcomes:

1. **PO1: Computational Knowledge:** Demonstrate competencies in fundamentals of computing, computing specialization, mathematics, and domain knowledge suitable for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
2. **PO2: Problem Analysis (PA):** Identify, formulate, and analyze complex real-life problems in order to arrive at computationally viable conclusions using fundamentals of mathematics, computer sciences, management and relevant domain disciplines.
3. **PO3: Design / Development of Solutions (DDS):** Design efficient solutions for complex, real-world problems to design systems, components or processes that meet the specifications with suitable consideration to public health, safety, cultural, societal, and environmental considerations.
4. **PO4: Conduct Investigations of Complex Computing Problems (CICP):** Ability to research, analyze and investigate complex computing problems through design of experiments, analysis and interpretation of data and synthesis of the information to arrive at valid conclusions.
5. **PO5: Modern Tool Usage (MTU):** Create, select, adapt, and apply appropriate technologies and tools to a wide range of computational activities while understanding their limitations.
6. **PO6: Professional Ethics (PE):** Ability to perform professional practices in an ethical way, keeping in mind cyber regulations & laws, responsibilities, and norms of professional computing practices.
7. **PO7: Life-long Learning (LLL):** Ability to engage in independent learning for continuous self- development as a computing professional.
8. **PO8: Project Management and Finance (PMF):** Ability to apply knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects in multidisciplinary environments.
9. **PO9: Communication Efficacy (CE):** Ability to effectively communicate with the technical community and with the society at large about complex computing activities by being able to understand and write effective reports, design documentation, make effective presentations with the capability of giving and taking clear instructions.

10. **PO10: Societal and Environmental Concern (SEC):** Ability to recognize and assess societal, environmental, health, safety, legal and cultural issues within local and global contexts and the consequential responsibilities applicable to professional computing practices.
11. **PO11: Individual and Teamwork (I&T):** Ability to work in multi-disciplinary team collaboration both as a member and leader, as per need.
12. **PO12: Innovation and Entrepreneurship (I&E):** Ability to apply innovation to track a suitable opportunity to create value and wealth for the betterment of the individual and society at large.

II. Relevance of MCA Programme in Chhatrapati Shahu Ji Maharaj University Kanpur's Mission and Objectives:

Master of Computer Application (MCA) program with the mission and objectives of Chhatrapati Shahu Ji Maharaj University, Kanpur, it's essential to consider how the program contributes to the university's overarching goals and values. Here's how the relevance of an MCA program could be articulated in relation to the mission and objectives of the university:

1. ***Promoting Access to Education:*** The MCA programme plays a crucial role in promoting access to quality education by offering flexible learning options, including distance and online education. This ensures that individuals from diverse backgrounds and locations, aspiring to pursue a career in the field of computer applications, can access high-quality education regardless of their geographical constraints or personal circumstances.
2. ***Preparing Students for Carriers and Leadership:*** The MCA programme is dedicated to preparing students for successful careers and leadership roles in the dynamic field of information technology. Through a well-rounded curriculum and a range of practical experiences, students are equipped with essential knowledge, skills, and competencies to excel in various sectors of the IT industry.
3. ***Emphasizing Research:*** The MCA programme prioritizes research, fostering critical thinking and intellectual curiosity among students and faculty. By engaging in research projects, students contribute to the advancement of knowledge in computer science and information technology, preparing them to be innovative problem solvers in the industry.

MCA program with the mission and objectives of Chhatrapati Shahu Ji Maharaj University, Kanpur, it not only enhances the relevance and effectiveness of the program but also strengthens the overall impact of the university in serving its stakeholders and society at large.

III. Nature of prospective target group of learners:

The prospective target group of learners for Master of Computer Application (MCA) program can vary depending on factors such as the program's focus, delivery mode, and institutional context. However, there are several common characteristics and attributes that are often associated with the typical demographic profile of MCA students:

1. ***University Graduates:*** The MCA programme appeals to students who have recently completed their graduation and are eager to pursue postgraduate studies in the field of computer application. These students typically possess a solid academic foundation and are driven by the desire to acquire a degree that will equip them with the necessary skills and knowledge to embark on a successful career in the IT industry or related fields.
2. ***Carrier Advancers:*** Prospective MCA students aim for careers in IT and computer science, including roles like software developer, systems analyst, or IT consultant. Some aspire to start tech start-ups, lead in top companies, or specialize in areas like cybersecurity or data science.
3. ***Motivated and Ambitious:*** MCA students are often characterized by their ambition,

motivation, and drive to succeed. They are willing to put in the effort required to excel academically and take advantage of opportunities for professional development and networking.

4. ***Diverse Backgrounds:*** MCA programs often attract students from diverse cultural, ethnic, and socioeconomic backgrounds. This diversity enriches the learning environment and provides students with opportunities to interact with peers from different perspectives and experiences.
5. ***Entrepreneurial Spirit:*** Some prospective MCA students may have an entrepreneurial spirit and aspirations to start their own businesses or ventures. They are interested in learning about business concepts, strategies, and practices that will help them succeed as entrepreneurs.
6. ***Economically Diverse Students:*** The program appeals to students from diverse socioeconomic backgrounds who seek affordable and accessible educational opportunities. These learners may appreciate programs that have flexible payment options to make education more accessible.
7. ***Skill Up-graders:*** Some prospective students may enroll in MCA-ODL program to upgrade their skills or transition to new career paths within IT fields. They may be looking to acquire advanced IT skills that are in demand in today's job market.
8. ***Specialized Learners:*** This program attracts students with specific interests or career goals within the IT field. These learners may seek programs that offer specialized tracks, concentrations, or elective courses tailored to their areas of interest.

IV. Appropriateness of program to be conducted in Open and Distance Learning mode to acquire specific skills and competence:

Conducting a Master of Computer Application (MCA) program in Open and Distance Learning (ODL) mode can be highly appropriate for acquiring specific skills and competencies, particularly for learners who require flexibility, accessibility, and personalized learning experiences. Here's why the ODL mode can be beneficial for acquiring skills and competence in MCA program:

1. ***Flexibility:*** ODL programs offer learners the flexibility to study at their own pace and convenience. This flexibility is particularly valuable for individuals who may have work commitments, family responsibilities, or other constraints that make attending traditional on-campus classes challenging. As a result, learners can balance their studies with other commitments, allowing them to acquire skills and competence in MCA program without disrupting their personal or professional lives.
2. ***Accessibility:*** ODL programs make education more accessible to a broader range of learners, including those who are geographically isolated or unable to attend traditional on-campus classes due to mobility issues or other barriers. By removing geographical constraints, ODL programs enable learners from diverse backgrounds and locations to participate in MCA program and acquire the skills and competence needed for success in the business world.
3. ***Personalized Learning:*** ODL programs often utilize technology-enabled learning platforms that allow for personalized learning experiences. Learners can access a variety of resources, including multimedia content, online lectures, discussion forums, and interactive simulations, tailored to their individual learning styles and preferences. This personalized approach can enhance engagement, comprehension, and retention of key concepts and skills in the MCA program.
4. ***Technology Integration:*** MCA programs conducted in ODL mode leverage technology to facilitate learning, collaboration, and communication among learners and instructors. Through online platforms, learners can engage in virtual classrooms, participate in group discussions, submit assignments, and receive feedback from instructors in real-time. This

integration of technology not only enhances the learning experience but also prepares learners for the digital workplace, where technology skills are increasingly essential.

5. ***Self-Directed Learning Skills:*** ODL programs promote the development of self-directed learning skills, including time management, organization, and self-motivation. Learners in MCA program conducted in ODL mode take greater responsibility for their learning journey, setting goals, managing their study schedules, and seeking out resources to enhance their skills and competence. These self-directed learning skills are highly valuable in the dynamic and rapidly changing business environment.
6. ***Cost Effectiveness:*** ODL programs often offer cost-effective alternatives to traditional on-campus education, as they eliminate the need for expenses such as commuting, accommodation, and campus facilities. This affordability makes acquiring skills and competence in MCA program more accessible to learners from diverse socioeconomic backgrounds, thereby promoting inclusivity and equity in education.

Overall, conducting MCA program in Open and Distance Learning mode can be highly appropriate for acquiring specific skills and competencies, offering flexibility, accessibility, personalized learning experiences, technology integration, self-directed learning skills, and cost-effectiveness. These advantages make ODL programs an attractive option for learners seeking to acquire business knowledge and skills while balancing their personal and professional commitments.

V. Instructional Design of Open and Distance Learning mode to acquire specific skills and competence:

Designing the instructional framework for an Open and Distance Learning (ODL) mode of Master of Computer Application (MCA) to acquire specific skills and competence requires careful consideration of various factors to ensure effectiveness, engagement, and learner success. Here's a structured approach to instructional design for such program:

A. Curriculum Design:

The curriculum of the MCA programme is meticulously designed with inputs from industry experts, Bloom's taxonomy, and faculty knowledge to offer students a comprehensive and contemporary education in computer applications. By integrating the latest industry insights and trends, the curriculum ensures students are well-prepared for the dynamic demands of the modern IT landscape. Employing Bloom's Taxonomy, the curriculum focuses on developing higher order thinking skills such as critical analysis, problem-solving, and evaluation, enabling students to tackle complex challenges with confidence. The expertise of faculty members enriches the curriculum, providing students with practical wisdom and industry insights. Through interactive lectures, hands-on projects, and engaging discussions, faculty members equip students with the tools needed to excel in their future careers. With a strong emphasis on practical learning and real-world applications, the MCA curriculum ensures students acquire the skills essential for success in today's competitive IT environment, bridging the gap between theory and practice to empower students to make meaningful contributions to the ever-evolving world of technology.

Semester-wise Titles of the Papers in MCA

MCA 1st Year (Semester I)

Year	Sem.	Course Code	Paper Title	Theory/ Practical	Credits
1st	I	MCA-1001	Fundamental of Computers & Emerging Technologies	Theory	4
1st	I	MCA-1002	Problem Solving using C	Theory	4
1st	I	MCA-1003	Principles of Management & Communication	Theory	4
1st	I	MCA-1004	Discrete Mathematics	Theory	4
1st	I	MCA-1005	Computer Organization & Architecture	Theory	4
1st	I	MCA-1051	Principles of Programming Using C Lab	Practical	3
1st	I	MCA-1052	Professional Communication Lab	Practical	2

MCA 1st Year (Semester II)

Year	Sem.	Course Code	Paper Title	Theory/ Practical	Credits
1st	II	MCA-2001	Theory of Automata & Formal Languages	Theory	4
1st	II	MCA-2002	Object Oriented Programming	Theory	4
1st	II	MCA-2003	Operating Systems	Theory	4
1st	II	MCA-2004	Database Management Systems	Theory	4
1st	II	MCA-2005	Data Structures & Analysis of Algorithms	Theory	4
1st	II	MCA-2051	DBMS Lab	Practical	3
1st	II	MCA-2052	Object oriented and data structure lab	Practical	3

MCA 2nd Year (Semester III)

Year	Sem.	Course Code	Paper Title	Theory/ Practical	Credits
2nd	III	MCA-3001	Computer Network	Theory	4
2nd	III	MCA-3002	Artificial Intelligence	Theory	4
2nd	III	MCA-3003	Software Engineering	Theory	4
2nd	III		Elective – 1	Theory	4
2nd	III		Elective – 2	Theory	4
2nd	III	MCA-3051	Software Engineering Lab	Practical	3
2nd	III	MCA-3052	Mini Project(AI / ISCL)	Practical	4

MCA 2nd Year (4th Semester)

Year	Sem.	Course Code	Paper Title	Theory/ Practical	Credits
2nd	IV		Elective – 3	Theory	4
2nd	IV		Elective – 4	Theory	4
2nd	IV		Elective – 5	Theory	4
2nd	IV	MCA-4061	Major Project	Practical	15

ELECTIVE SUBJECTS		
Elective-1	MCA-3004	Data Warehousing & Data Mining
	MCA-3005	Cloud Computing
Elective-2	MCA-3006	Big Data
	MCA-3007	Digital Image Processing
Elective-3	MCA-4001	Soft Computing
	MCA-4002	Software Quality Engineering
Elective-4	MCA-4003	Neural Network
	MCA-4004	Internet of Things
Elective-5	MCA-4005	Machine Learning
	MCA-4006	Quantum Computing

B. Detailed Syllabus: Annexure – I

C. Duration of the Programme: 02 Years - divided into 04 semesters.

D. Faculty and Support Staff requirement:

Academic Staff

1-Programme Coordinator, 1- Course Coordinator, 1-Course Mentor per batch of 50 students

E. Instructional Delivery mechanisms & Identification of Media:

The methodology of instruction in this course will be different from that of the other conventional (regular/ physical) courses run in the University. A student-centric and student-convenient approach is required in the open and distance learning (ODL) courses. This is also important because learning/ instruction is imparted through print and/ or audio-visual media rather than face-to-face communication.

F. Self-Learning Materials (SLM) should be developed in print media:

- Self-Learning Materials (SLM), in print media, shall be developed.
- SLM would be self-explanatory, self-contained, self-directed, self-motivating and self-evaluating.
- There shall be a description of the credit value of each module or unit in the course.
- There shall be clear guidelines on academic integrity and netiquette (internet etiquette) expectations regarding activities, discussions, and plagiarism.
- The audio-visual material will supplement and complement the Self Learning Materials and will be based on the curriculum structure.
- The level and style of presentation and language should be simple and appropriate to facilitate e-learning.
- The content must be interactive with the appropriate use of graphics, animation simulations, etc. to keep students interested.

G. Student support service systems:

The main goal of student support service systems is to promote independent or independent study. Study among distance learners in the absence of regular face-to-face teaching. All the time educational support will be provided to students. Support will be available all the time in the following areas:

- Information, tips and advice about the programme.
- Advice before admission, during admission, and after admission.
- Introduction for new students.
- Provide academic advising schedules and practice schedules.
- Evaluate students and exchange feedback.
- Support with other academic and administrative inquiries such as registration and examination Rating, comments, etc**.**

VI. Procedure for Admissions, Curriculum Transaction and Evaluation:

The purpose of online and distance education is to provide flexible learning opportunities to students to attain qualification, wherever learners are not able to attend the regular classroom teaching. The programme is called online mode for the award of Degree.

A. Procedure for Admission

Relevant undergraduate program from a recognized University. Candidate must have passed Mathematics at 10+2 level and/ or graduation level.

B. Curriculum Transaction and Evaluation

The marking is divided into two parts:

a. For continuous internal assessment (CIA) through projects and assignment writings, and

b. For end semester evaluation through offline examination.

VII. Library Resources:

Online Study Material and its availability is one most identified concern for the students to have access to online course material and resources.

VIII. Cost estimate of the program and the provisions:

Suggested Fee for MCA-ODL is as per the CSJM University norms (This fee includes Self Learning Material cost, Learning Management System maintenance cost and Subject Matter Expert cost).

IX. Quality Assurance Mechanism and Programme Learning Outcomes:

A. Quality Assurance Mechanism:

MCA-ODL program is agreed to the latest pedagogies and prepares you for many contours your professional life might take.

The key points which make our offered programme much better in terms evaluation criteria:

- The programme is being offered by NAAC A++ ranked Chhatrapati Shahu Ji Maharaj University, Kanpur.
- Highly qualified faculty who bring professional experience into the classroom.
- Relevant courses are immediately applicable to the workplace.
- Dedicated student support services.
- Flexible ways to learn.

B. Program Learning Outcomes:

1. To be able to understand problems, think of the best suitable approach to solve the problem, develop, and evaluate effective solutions as per the local/regional/ national/ global requirements and availability of resources/ technologies.
2. To be able excel in contemporary technologies being adopted by the industry and academia for providing sustainable solutions.
3. To be able to excel in various programming/project competitions and technological challenges laid by professional bodies.

Annexure – I

<table>
<tr><th colspan="5">MCA Semester I, Paper-I (04 Credits)</th></tr>
<tr><th colspan="5">Core Course: MCA-1001 FUNDAMENTAL OF COMPUTERS & EMERGING TECHNOLOGIES</th></tr>
<tr><th colspan="2">Credit: 4</th><th>CIA: 25</th><th>ESE: 75</th><th>Max. Marks: 100</th></tr>
<tr><td colspan="5">This course aims to provide a comprehensive understanding of computer fundamentals, covering essential concepts such as hardware, software, operating systems, and basic programming principles. Students will develop proficiency in navigating computer systems, troubleshooting common issues, and utilizing productivity tools effectively. By the end of the course, learners will possess the foundational knowledge required to make informed decisions about technology use, enhance digital literacy skills, and lay the groundwork for further studies or professional endeavors in computing.</td></tr>
<tr><td>Block I</td><td colspan="3">Unit 1: Introduction to Computer: Definition, Computer Hardware & Computer Software
Unit 2: Components: Hardware – Introduction, Input devices, Output devices, Central Processing Unit, Memory-Primary and Secondary, Software Introduction, Types – System and Application.
Unit 3: Computer Languages: Introduction, Concept of Compiler, Interpreter & Assembler
Unit 4: Problem solving concept: Algorithms – Introduction, Definition, Characteristics, Limitations, Conditions in pseudo-code, Loops in pseudo code.</td><td></td></tr>
<tr><td>Block II</td><td colspan="3">Unit 1: Operating system: Definition, Functions, Types, Classification,
Unit 2: Elements of command based, and GUI based operating system.
Unit 3: Computer Network: Overview, Types (LAN, WAN and MAN),
Unit 4: Data communication topologies</td><td></td></tr>
<tr><td>Block III</td><td colspan="3">Unit 1: Architecture, Functioning, Basic services like WWW
Unit 2: FTP, Telnet, Gopher etc. Search engines, E-mail, Web Browsers
Unit 3: Internet of Things(IoT): Definition, Sensors, their types, and features,
Unit 4: Smart Cities, Industrial Internet of Things</td><td></td></tr>
<tr><td>Block IV</td><td colspan="3">Unit 1: Introduction, overview, features, limitations,
Unit 2: Application areas, fundamentals of Block Chain
Unit 3: Introduction, Applications and use cases.
Unit 4: IT nature and benefits, AWS, Google, Microsoft & IBM Services,</td><td></td></tr>
<tr><td>Block V</td><td colspan="3">Unit 1: Emerging Technologies: Introduction, overview, features, limitations
Unit 2: Application areas of Augmented Reality,
Unit 3: Virtual Reality, Grid computing, Green computing
Unit 4: Big data analytics, Quantum Computing and Brain Computer Interface.</td><td></td></tr>
</table>

Suggested Readings:

1. Rajaraman V., "*Fundamentals of Computers*", Prentice-Hall of India
2. Norton P., "*Introduction to Computers*", Mc Graw Hill Education.
3. Goel A., "*Computer Fundamentals*", Pearson.
4. Balagurusamy E., "*Fundamentals of Computers*", Mc Graw Hill
5. Thareja R., "*Fundamentals of Computers*", Oxford University Press

MCA Semester I, Paper-II (04 Credits)			
Core Course: MCA-1002 PROBLEM SOLVING USING C			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to cultivate problem-solving skills using the C programming language. Through hands-on exercises and projects, students will learn algorithmic thinking, data structures, and procedural programming concepts. By mastering fundamental programming techniques, students will develop the ability to analyze problems, devise efficient solutions, and implement them using C. Emphasis will be placed on understanding program flow, debugging strategies, and optimizing code

Block I	**Unit 1:** Basics of programming**:** Approaches to problem solving, Use of high-level programming **Unit 2:** Concept of algorithm and flowchart, Concept, and role of structured programming. **Unit 3:** Basics of C: History of C, Salient features of C, Structure of C Program, Compiling C Program, Link and Run C Program, **Unit 4:** Character set, Tokens, Keywords, Identifiers, Constants, Variables, Instructions, Data types, Standard Input/Output, Operators, and expressions.	
Block II	**Unit 1:** Conditional Program Execution: if, if-else, and nested if-else statements, Switch statements, **Unit 2:** Restrictions on switch values, Use of break and default with switch, Comparison of switch and if-else. **Unit 3:** Loops and Iteration: for, while and do-while loops, Multiple loop variables, Nested loops, Assignment operators, break and continue statement. **Unit 4:** Functions: Introduction, Types, Declaration of a Function, Function calls, Defining functions, Function Prototypes, Passing arguments to a function Return values and their types, Writing multifunction program, Calling function by value, Recursive functions.	
Block III	**Unit 1:** Arrays: Array notation and representation, Declaring one-dimensional array, Initializing arrays, Accessing array elements, Manipulating array elements, **Unit 2:** Pointers: Introduction, Characteristics, * and & operators, Pointer type declaration and assignment, Pointer arithmetic, **Unit 3:** Call by reference, Passing pointers to functions, array of pointers, Pointers to functions, Pointer to pointer, Array of pointers. **Unit 4: Strings:** Introduction, Initializing strings, Accessing string elements, Array of strings, Passing strings to functions, String functions.	
Block IV	**Unit 1: Structure:** Introduction, Initializing, defining and declaring structure, Accessing members, Operations on individual members, Operations on structure **Unit 2:** Structure within structure, Array of structure, Pointers to structure. **Unit 3:** Union: Introduction, Declaring union, Usage of unions, Operations on union. Enumerated data types **Unit4:** Storage classes: Introduction, Types- automatic, register, static and external.	
Block V	**Unit 1:** Dynamic Memory Allocation: Introduction, Library functions – malloc, calloc, realloc and free. **Unit 2:** File Handling: Basics, File types, File operations, File pointer, File opening modes, File handling functions, **Unit 3:** File handling through command line argument, Record I/O in files. **Unit 4:** Graphics: Introduction, Constant, Data types and global variables used in graphics, Library functions used indrawing, Drawing and filling images, GUI interaction within the program.	

Suggested Readings:

1. Kanetkar Y., "*Let Us C*", BPB Publications
2. Hanly J. R. and Koffman E. B., "*Problem Solving and Program Design in C*", Pearson Education.
3. Schildt H., "*C - The Complete Reference*", McGraw-Hill.
4. Goyal K. K. and Pandey H. M., "*Trouble Free C*", University Science Press

MCA Semester I, Paper-III (04 Credits)			
Core Course: MCA-1003 PRINCIPLES OF MANAGEMENT & COMMUNICATION			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to equip students with essential principles of management and effective communication strategies. Students will explore foundational management concepts including planning, organizing, leading, and controlling, alongside the importance of interpersonal communication in organizational settings. Through case studies, simulations, and practical exercises, students will develop critical thinking skills, leadership qualities, and the ability to communicate persuasively and professionally.

Block I	**Unit 1:** Management: Need, Scope, Meaning and Definition. The process of Management, **Unit 2:** Development of Management thought F.W. Taylor and Henry Fayol, **Unit 3:** Horothorne Studies, Qualities of an Efficient Management.	
Block II	**Unit 1:** Planning & Organizing: Need, Scope and Importance of Planning, Steps in planning, **Unit 2:** Decision making model. Organizing need and Importance, Organizational Design, **Unit 3:** Organizational structure, centralization and Decentralization, Delegation.	
Block III	**Unit 1:** Directing & Controlling: Motivation—Meaning, Importance, need. Theories of Motivation, **Unit 2:** Leadership—meaning, need and importance, leadership style, Qualities of effective leader, **Unit 3:** principles of directing, Basic control process, Different control Techniques.	
Block IV	**Unit 1:** Introduction to Communication: What is Communication, Levels of communication, Barriers to communication, Process of Communication, Non-verbal Communication, **Unit 2:** The flow of Communication: Downward, Upward, Lateral or Horizontal (Peer group) Communication, Technology Enabled communication, Impact of Technology, **Unit 3:** Selection of appropriate communication Technology, Importance of Technical communication.	
Block V	**Unit 1:** Business letters: Sales & Credit letters; Claim and Adjustment Letters; Job application and Resumes. **Unit 2:** Reports: Types; Structure, Style & Writing of Reports. **Unit 3:** Technical Proposal**:** Parts; Types; Writing of Proposal; Significance. Nuances of Delivery; Body Language; Dimensions of Speech: Syllable; Accent; Pitch; Rhythm; Intonation. **Unit 4:** Paralinguistic features of voice; Communication skills, Presentation strategies, Group Discussion; Interview skills; Workshop; Conference; Seminars.	

Suggested Readings:

1. P. C. Tripathi, P. N. Reddy, "*Principles of Management*", McGraw Hill Education 6th Edition.
2. C. B. Gupta, "*Management Principles and Practice*", Sultan Chand & Sons 3rd edition.
3. T. N. Chhabra, "*Business Communication*", Sun India Publication.
4. V. N. Arora and Laxmi Chandra, "*Improve Your Writing*", Oxford Univ. Press, 2001, New Delhi.
5. Madhu Rani and Seema Verma, "*Technical Communication: A Practical Approach*", Acme Learning, New Delhi-2011.
6. Meenakshi Raman & Sangeeta Sharma, "*Technical Communication Principles and Practices*", Oxford Univ. Press, 2007, New Delhi.
7. Koontz Harold & Weihrich Heinz, "*Essentials of Management*", McGraw Hill 5thEdition 2008.
8. Robbins and Coulter, "*Management*", Prentice Hall of India, 9th edition.
9. James A. F., Stoner, "*Management*", Pearson Education Delhi.

MCA Semester I, Paper-IV (04 Credits)			
Core Course: MCA 1004 DISCRETE MATHEMATICS			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to provide a rigorous foundation in discrete mathematics, focusing on fundamental concepts such as sets, logic, relations, functions, and combinatorics. Students will develop analytical thinking skills essential for computer science and mathematics applications. Topics include mathematical reasoning, proof techniques, graph theory, and discrete probability

Block I	**Unit 1:** Set Theory: Introduction, Size of sets and Cardinals, Venn diagrams, Combination of sets, Multisets, Ordered pairs and Set Identities. **Unit 2:** Relation: Definition, Operations on relations, Composite relations, Properties of relations, Equality of relations, Partial order relation. **Unit 3:** Functions: Definition, Classification of functions, Operations on functions, Recursively defined functions.	
Block II	**Unit 1:** Posets, Hasse Diagram and Lattices: Introduction, Partial ordered sets, Combination of Partial ordered sets, **Unit 2:** Hasse diagram, Introduction of lattices, Properties of lattices – Bounded, Complemented, Modular and Complete lattice. **Unit 3:** Boolean Algebra: Introduction, Axioms and Theorems of Boolean algebra, Boolean functions. Simplification of Boolean functions, Karnaugh maps, Logic gates. **Unit 4:** Propositional: Propositions, Truth tables, Tautology, Contradiction, Algebra of Propositions, Theory of Inference and Natural Detection.	
Block III	**Unit 1:** Algebraic Structures: Introduction to algebraic Structures and properties. Types of algebraic structures: Semi group, Monoid, Group, **Unit 2:** Abelian group and Properties of group. Subgroup, Cyclic group, Cosets, Permutation groups, Homomorphism, and Isomorphism of groups. **Unit 3:** Rings and Fields**:** Definition and elementary properties of Rings and Fields	
Block IV	**Unit 1:** Natural Numbers: Introduction, Piano's axioms, Mathematical Induction, Strong Induction, and Induction with Nonzero Base cases. **Unit 2:** Recurrence Relation & Generating functions: Introduction and properties of Generating Functions. Simple Recurrence relation with constant coefficients and Linear recurrence relation without constant coefficients. **Unit 3:** Methods of solving recurrences. Combinatorics: Introduction, Counting techniques and Pigeonhole principle, Polya's Counting theorem.	
Block V	**Unit 1:** Graph theory: Path, cycles, handshaking theorem, bipartite graphs, sub-graphs, graph isomorphism, operations on graphs, Eulerian graphs, and Hamiltonian graphs, **Unit 2:** planar graphs, Euler formula, traveling salesman problem, shortest path algorithms. **Unit 3:** Euler tours, planar graphs, Euler's formula, applications of Kuratowski's theorem, **Unit 4:** graph coloring, chromatic polynomials, trees, weighted trees, shortest path algorithms, spanning trees.	

Suggested Readings:

1. Kenneth H. Rosen, "*Discrete Mathematics and Its Applications*", McGraw Hill, 2006.
2. B. Kolman, R.C Busby and S.C Ross, "*Discrete Mathematics Structures*", Prentice Hall, 2004.
3. R. P Grimaldi, "*Discrete and Combinatorial Mathematics*", Addison Wesley, 2004.
4. Y. N. Singh, "*Discrete Mathematical Structures*", Wiley-India, First edition, 2010.
5. Swapan Kumar Sarkar, "*A Textbook of Discrete Mathematics*", S. Chand & Company PVT. LTD.
6. V. Krishnamurthy, "*Combinatorics Theory & Application*", East-West Press Pvt. Ltd., New Delhi.
7. Lipschutz, Seymour, "*Discrete Mathematics*", McGraw Hill.

MCA Semester I, Paper-V (04 Credits)			
Core Course: MCA-1005 COMPUTER ORGANIZATION & ARCHITECTURE			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to provide a comprehensive understanding of computer organization and architecture principles. Students will explore the structure and function of digital computer systems, including CPU design, memory hierarchy, input/output systems, and assembly language programming. Emphasis will be placed on the interaction between hardware and software components, as well as performance optimization techniques.

Block I	**Unit 1:** Introduction: Functional units of digital system and their interconnections, **Unit 2:** buses, bus architecture, types of buses and bus arbitration. Register, bus and memory transfer. **Unit 3:** Processor organization: general registers organization, stack organization and addressing modes.	
Block II	**Unit 1:** Arithmetic and logic unit: Look ahead carries adders. Multiplication: Signed operand multiplication, **Unit 2:** Booth's algorithm and array multiplier. Division and logic operations. **Unit 3:** Floating point arithmetic operation, Arithmetic & logic unit design. IEEE Standard for Floating Point Numbers.	
Block III	**Unit 1:** Control Unit: Instruction types, formats, instruction cycles and sub cycles (fetch and execute etc.), **Unit 2:** micro-operations, execution of a complete instruction. Program Control, Reduced Instruction Set Computer, Pipelining. **Unit 3:** Hardwire and micro programmed control: micro-program sequencing, **Unit 4:** concept of horizontal and vertical microprogramming.	
Block IV	**Unit 1:** Memory:BasicconceptandhierarchySemiconductorRAMmemories,2D **Unit 2:** memory organization. ROM memories. Cache memories: concept and designs uses & performance, **Unit 3:** address mapping and replacement Auxiliary memories: magnetic disk **Unit 4:** magnetic tape and optical disks Virtual memory: concept implementation.	
Block V	**Unit 1:** Input / Output: Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. **Unit 2:** Modes of Data Transfer: Programmed I/O, interrupt initiated I/O and Direct Memory Access., I/O channels and processors. **Unit 3:** Serial Communication: Synchronous & asynchronous communication, standard communication interfaces.	

Suggested Readings:

1. John P. Hayes, "*Computer Architecture and Organization*", McGraw-Hill.
2. William Stallings, "*Computer Organization and Architecture-Designing for Performance*", Pearson Education.
3. M. Morris Mano, "*Computer System Architecture*", PHI.
4. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "*Computer Organization*", McGraw-Hill.
5. Behrooz Parahami, "*Computer Architecture*", Oxford University Press.
6. David A. Patterson and John L. Hennessy, "*Computer Architecture - A Quantitative Approach*", Elsevier Pub.
7. Tannenbaum, "*Structured Computer Organization*", PH

MCA Semester II, Paper-I (04 Credits)			
Core Course : MCA-2001 THEORY OF AUTOMATA & FORMAL LANGUAGES			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
Automata theory explores abstract computational models like finite automata and Turing machines. Formal languages are sets of strings defined by rules. This theory underpins computer science, aiding in language processing, compiler design, and algorithm analysis by providing tools to understand computation and recognize patterns within strings.			

Block I	**Unit 1:** Introduction to Theory of Computation-Automata, Computability and Complexity, Alphabet, Symbol, String, Formal Languages, Deterministic **Unit 2:** Finite Automaton(DFA)-Definition, Representation, Acceptability of a String and Language, Non-Deterministic **Unit 3:** Finite Automaton (NFA), Equivalence of DFA and NFA, NFA with ε-Transition, Equivalence of NFA's with and without ε-Transition, Finite Automata with output-Moore machine, **Unit 4:** Mealy Machine, Equivalence of Moore and Mealy Machine, Minimization of Finite Automata, Myhill-Nerode Theorem, Simulation of DFA and NFA.	
Block II	**Unit 1:** Regular Expressions and Languages**:** Regular Expressions, Transition **Unit 2:** Graph, Kleen's Theorem, Finite Automata and Regular Expression-Arden's theorem, Algebraic Method Using Arden's Theorem, Regular and Non-Regular Languages **Unit 3:** Closure properties of Regular Languages, Pigeonhole Principle, Pumping Lemma, Application of Pumping Lemma, **Unit 4:** Decidability-Decision properties, Finite Automata and Regular Languages, Regular Languages and Computers, Simulation of Transition Graph and Regular language.	
Block III	**Unit 1:** Context Free Grammar (CFG)-Definition, Derivations, Languages, **Unit 2:** Derivation Trees and Ambiguity, Regular Grammars-Right Linear and Left Linear grammars, Conversion of FA into CFG and Regular grammar into **Unit 3:** FA, Simplification of CFG, Normal Forms-Chomsky Normal Form (CNF), Greibach Normal Form (GNF), Chomsky **Unit 4:** Hierarchy, Programming problems based on the properties of CFGs.	
Block IV	**Unit 1:** Push Down Automata and Properties of Context Free Languages: **Unit 2:** Nondeterministic Pushdown Automata (NPDA)-Definition, Moves, A Language Accepted by NPDA, Deterministic Pushdown Automata (DPDA) and Deterministic Context free Languages (DCFL), **Unit 3:** Pushdown Automata for Context Free Languages, Context Free grammars for Pushdown Automata, Two stack Pushdown Automata, **Unit 4:** Pumping Lemma for CFL, Closure properties of CFL, Decision Problems of CFL, Programming problems based on the properties of CFLs.	
Block V	**Unit 1:** Turing Machines and Recursive Function Theory: Basic Turing Machine Model, Representation of Turing Machines **Unit 2:** Language Acceptability of Turing Machines, Techniques for Turing Machine Construction, Modifications of Turing Machine, Turing Machine as Computer of Integer **Unit 3:** Functions, Universal Turing machine, Linear Bounded Automata, Church's Thesis, Recursive and Recursively **Unit 4:** Enumerable language, Halting Problem Post Correspondence Problem, Introduction to Recursive Function Theory.	

Suggested Readings:

1. J. E. Hopcraft, R. Motwani, and Ullman, "*Introduction to Automata theory, Languages and Computation*", Pearson Education Asia, 2nd Edition.
2. J. Martin, "*Introduction to languages and the theory of computation*", McGraw Hill, 3rd Edition.
3. C. Papadimitriou and C. L. Lewis, "*Elements and Theory of Computation*", PHI.
4. K. L. P. Mishra and N. Chandrasekaran, "*Theory of Computer Science Automata Languages and Computation*" PHI.
5. Y. N. Singh, "*Mathematical Foundation of Computer Science*", New Age International.

MCA Semester II, Paper-II (04 Credits)			
Core Course: MCA-2002 OBJECT ORIENTED PROGRAMMING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

Object-oriented programming (OOP) is a programming paradigm where programs are organized around objects rather than actions or logic. Objects encapsulate data and behaviour, communicating through methods. OOP promotes code reusability, modularity, and flexibility, facilitating easier maintenance and development of complex software systems. Common OOP languages include Java, Python, and C++.

Block I	**Unit 1:** Object Oriented Programming: objects, classes, Abstraction, Encapsulation, Inheritance, Polymorphism, OOP in Java, Characteristics of Java, **Unit 2:** The Java Environment, Java Source File Structure, and Compilation. **Unit 3:** Fundamental Programming Structures in Java **Unit 4:** Defining classes in Java, constructors, methods, access specifies, static members, Comments, Data Types, Variables, Operators, Control Flow, Arrays.	
Block II	**Unit 1:** Inheritance, Interfaces, and Packages: Inheritance: Super classes, subclasses, Protected members, constructors in subclasses, Object class, abstract classes, and methods. **Unit 2:** Interfaces: defining an interface implementing interface, differences between classes and interfaces and extending interfaces, Object cloning, inner classes. Packages: Defining Package, CLASSPATH Setting for Packages, **Unit 3:** Making JAR Files for Library Packages, Import and Static Import Naming Convention for Packages, Networking java.net package.	
Block III	**Unit 1:** Exception Handling, I/O: Exceptions: exception hierarchy, throwing and catching exceptions, built. **Unit 2:**in exceptions, creating own exceptions, Stack Trace Elements. Input **Unit 3:** Output Basics: Byte streams and Character streams, Reading and Writing, Console Reading, and Writing Files.	
Block IV	**Unit 1: Multithreading and Generic Programming:** Differences between multithreading **Unit 2**: and multitasking, thread lifecycle, creating threads, synchronizing threads, Interthread, read communication, daemon threads, thread groups. **Unit 3:** Generic Programming: Generic classes, generic methods, **Unit4:**BoundedTypes:RestrictionsandLimitations.	
Block V	**Unit 1: Event Driven Programming:** Graphics programming: Frame, Components, workingwith2Dshapes,Usingcolors,fonts,and images. Basics of event handling: event **Unit 2:** handlers, adapter classes, actions, mouse events, AWT even their AWT event hierarchy Introduction to Swing: layout management, Swing **Unit 3: C**omponents: Text Fields, Text Areas, Buttons, Check Boxes, Radio Buttons, Lists, choices, Scrollbars, **Unit 4:** Windows Menus and Dialog Boxes.	

Suggested Readings:

1. Herbert Schildt, "*Java The complete reference*", McGraw-Hill Education, 8th Edition, 2011
2. Cay S. Horstmann, Gary Cornell, "*Core Java Volume-I Fundamentals*", Prentice Hall, 9th Edition, 2013.
3. Steven Holzner, "*Java Black Book*", Dreamtech.
4. Balagurusamy E, "*Programming in Java*", McGraw-Hill
5. Naughton, Schildt, "*The Complete Reference Java 2*", McGraw Hill
6. Khalid Mughal, "*A Programmer's Guide to Java SE8*" Oracle Certified Associate (OCA), Addison-Wesley.

MCA Semester II, Paper-III (04 Credits)			
Core Course: MCA-2003 OPERATING SYSTEMS			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

An operating system (OS) manages computer hardware and software resources, providing services to software applications. It controls memory, scheduling tasks, handling input/output operations, and facilitating communication between hardware components. Examples include Windows, macOS, and Linux. OS ensures efficient and secure utilization of computer resources, enabling user interaction and application execution.

Block I	**Unit 1:** Operating System Structure-Layered structure, System Components, **Unit 2:** Operating system functions, Classification of Operating systems-Batch, Interactive, Time-sharing, Real-Time System, Multiprocessor Systems, **Unit 3:** Multiuser Systems, Multi-process Systems, Multithreaded Systems, **Unit 4:** Operating System services, Reentrant Kernels, Monolithic and Microkernel Systems.	
Block II	**Unit 1: Concurrent Processes**: Process Concept, Principle of Concurrency, Producer/Consumer Problem **Unit 2:** Mutual Exclusion, Critical Section Problem, Dekker's solution, Peterson's solution, Semaphores, **Unit 3:** Test and Set operation, Classical Problem in Concurrency-Dining Philosopher Problem, Sleeping Barber Problem, **Unit 4:** Inter Process Communication models and Schemes Process generation.	
Block III	**Unit 1:** Scheduling Concepts, Performance Criteria, Process States, Process Transition Diagram, Schedulers, **Unit 2:** Process Control Block (PCB), Process address space, Process identification information, **Unit 3:** Threads and their management, Scheduling Algorithms, Multiprocessor Scheduling. **Unit 4:** Deadlock: System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock.	
Block IV	**Unit 1:** Basic bare machine, Resident monitor, Multiprogramming with fixed partitions **Unit 2:** Multiprogramming with variable partitions, Protection schemes, Paging, **Unit 3:** Paged segmentation, Virtual memory concepts, Demand paging, Performance of demand paging, Page replacement algorithms, **Unit 4:** Thrashing, Cache memory organization, Locality of reference.	
Block V	**Unit 1: /O Management and Disk Scheduling**: I/O devices, and I/O subsystems, I/O buffering, Disk storage and disk scheduling, RAID. **Unit 2:** File System: File concept, File organization and access mechanism, **Unit 3:** File directories, and File sharing, File system implementation issues, File system protection and security.	

Suggested Readings:

1. Silberschatz, Galvin and Gagne, "*Operating Systems Concepts*", Wiley Publication.
2. Sibsankar Halder and Alex A Arvind, "*Operating Systems*", Pearson Education.
3. Harvey M Dietel, "*An Introduction to Operating System*", Pearson Education.
4. William Stallings, "*Operating Systems: Internals and Design Principles*", 6th Edition, Pearson Education.
5. Harris, Schaum's Outline of "*Operating Systems*", McGraw Hill

MCA Semester II, Paper-IV (04 Credits)			
Core Course: MCA-2004 DATABASE MANAGEMENT SYSTEMS			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
Database Management Systems (DBMS) organize and store data, allowing users to retrieve, update, and manage information efficiently. They provide features for data integrity, security, and concurrency control. Examples include MySQL, Oracle, and PostgreSQL. DBMS ensures data consistency, enables data sharing, and supports complex queries for data analysis and decision-making.			
Block I	**Unit 1:** Overview Database System vs File System Database System Concept and Architecture Data Model Schema and Instances **Unit 2:** Data Independence and Database Language and Interfaces, Data Definitions Language, DML, Overall Database Structure. Data Modeling Using the Entity Relationship Model: ER Model Concept **Unit 3:** Notation for ER Diagram, Mapping Constraints, Keys, Concepts of Super Key, Candidate Key, **Unit 4:** Primary Key, Generalization, Aggregation, Reduction of an ER Diagrams to Tables, Extended ER Model, Relationship of Higher Degree.		
Block II	**Unit 1:** Relational Data Model Concepts, Integrity Constraints, Entity Integrity, Referential Integrity, Keys Constraints, Domain Constraints, Relational Algebra, Relational Calculus, Tuple and Domain Calculus. **Unit2:** Introduction to SQL Characteristics of SQL, Advantage of SQL. SQL Data type and Literal Types of SQL Commands. **Unit 3:** SQL Operators and their Procedure Tables, Views and Indexes Queries and Subqueries. **Unit 4:** Aggregate Functions Insert, Update and Delete Operations, Joins, Unions, Intersection, Minus, Cursors, Triggers, Procedures in SQL/PLSQL		
Block III	**Unit 1: Data Base Design & Normalization:** Functional dependencies, normal forms, **Unit 2:** first, second, third normal forms, BCNF, inclusion dependence, loss less join decompositions, **Unit 3:** normalization using FD, MVD, and JDs, alternative approaches to database design		
Block IV	**Unit 1: Transaction Processing Concept:** Transaction System, Testing of Serializability, Serializability of Schedules **Unit 2:** Conflict & View Serializable Schedule, Recoverability, Recovery from Transaction Failures, **Unit 3:** Log Based Recovery, Checkpoints, Deadlock Handling. **Unit 4:** Distributed Database: Distributed Data Storage, Concurrency Control, Directory System		
Block V	**Unit 1:** Concurrency Control, Locking Techniques for Concurrency Control **Unit 2:** Time Stamping Protocols for Concurrency Control, Validation Based Protocol **Unit 3:** Multiple Granularity, Multi Version Schemes, Recovery with Concurrent Transaction, Case Study of Oracle.		

Suggested Readings:

1. Korth, Silbertz, Sudarshan, "*Database Concepts*", McGraw Hill.
2. Date C J, "*An Introduction to Database Systems*", Addison Wesley.
3. Elmasri Navathe, "*Fundamentals of Database Systems*", Addison Wesley.
4. O'Neil, "*Databases*", Elsevier Pub.
5. Ramakrishnan, "*Database Management Systems*", McGraw Hill.
6. Leon & Leon, "*Database Management Systems*", Vikas Publishing House.
7. Bipin C. Desai, "*An Introduction to Database Systems*", Galgotia Publications.
8. Majumdar & Bhattacharya, "*Database Management System*", McGraw Hill.

MCA Semester II, Paper-V (04 Credits)			
Core Course: MCA-2005 DATA STRUCTURES & ANALYSIS OF ALGORITHMS			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
Data Structures & Analysis of Algorithms involves studying efficient ways to organize and manipulate data. Key concepts include arrays, linked lists, trees, graphs, and hash tables. Algorithm analysis evaluates efficiency in terms of time and space complexity. This field is fundamental in designing and optimizing software solutions for various computational problems.			
Block I	**Unit 1: Introduction to data structure:** Data, Entity, Information, Difference between Data and Information, Datatype, Building datatype, Abstract datatype, Definition of data structures, Types of Data Structures: Linear and Non-Linear Data Structure, **Unit 2: Introduction to Algorithms:** Definition of Algorithms, Difference between algorithm and programs, properties of algorithm, Algorithm Design Techniques, Performance Analysis of Algorithms, Complexity of various code structures, Order of Growth, Asymptotic Notations. **Unit 3: Arrays:** Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Derivation of Index Formulae for 1-D, 2-D Array Application of arrays, Sparse Matrices and their representations. **Unit 4: Linked lists:** Array Implementation and Pointer Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List, Insertion, Deletion, Traversal, Polynomial Representation and Addition Subtraction & Multiplications of Single variable.	
Block II	**Unit 1: Stacks**: Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Iteration and Recursion- Principles of recursion, Tail recursion, Removal of recursion Problem solving using iteration and recursion with examples such as binary search, Fibonacci numbers, and Hanoi towers. **Unit 2: Queues:** Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array, and linked implementation of queues in C, DE queue, and Priority Queue. **Unit 3: Searching:** Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & Collision resolution Techniques used in Hashing.	
Block III	**Unit 1: Sorting:** Insertion Sort, Selection Sort, Bubble Sort, Heap Sort, Comparison of Sorting Algorithms, Sorting in Linear Time: Counting Sort and Bucket Sort. **Unit 2: Graphs:** Terminology used with Graph, Data Structure for Graph Representations: Adjacency Matrices, Adjacency List, Adjacency. **Unit 3:** Graph Traversal: Depth First Search and Breadth First Search, Connected Component.	
Block IV	**Unit 1:** Basic terminology used with Tree, Binary Trees, Binary Tree Representation: Array Representation and Pointer (LinkedList) **Unit 2:** Representation, Binary Search Tree, Complete Binary Tree, An Extended Binary Trees **Unit 3:** Tree Traversal algorithms: In-order, Preorder and Post-order, Constructing Binary Tree from given Tree Traversal, **Unit 4:** Operation of Insertion, Deletion, Searching & Modification of data in Binary Search Tree, Threaded Binary trees, Huffman coding using Binary Tree, AVL Tree and B-Tree.	
Block V	**Unit 1:** Divide and Conquer with Examples Such as Merge Sort, Quick-Sort, **Unit 2:** Matrix Multiplication: Strassen's Algorithm Dynamic Programming: Dijkstra Algorithm, Bellman Ford Algorithm **Unit 3:** All-pair Shortest Path: Warshal's Algorithm, Longest Common Sub-Sequence Greedy Programming: Prims and Kruskal algorithm.	

Suggested Readings:

1. Cormen T. H., Leiserson C. E., Rivest R. L., and Stein C., "*Introduction to Algorithms*", PHI.
2. Horowitz Ellis, Sahni Sartaj and Rajasekharan S., "*Fundamentals of Computer Algorithms*", 2nd Edition, Universities Press.
3. Dave P.H., H. B. Dave, "*Design and Analysis of Algorithms*", 2nd Edition, Pearson Education.
4. Lipschitz S., "*Theory and Problems of Data Structures*", Schaum's Series.
5. Goyal K. K., Sharma Sandeep & Gupta Atul, "*Data Structures and Analysis of Algorithms*", HP Hamilton.
6. Lipschutz, "*Data Structures with C*", SIE-SOS, McGraw Hill
7. Samanta D, "*Classic Data Structures*", 2nd Edition Prentice Hall India.
8. Goodrich M.T. and Tomassia R., "*Algorithm Design: Foundations, Analysis and Internet examples*", John Wiley and sons.
9. Sridhar S., "*Design and Analysis of Algorithms*", Oxford Univ. Press.
10. Aho, Ullman and Hopcroft, "*Design and Analysis of algorithms*", Pearson Education.
11. R. Neapolitan and K. Naimipour, "*Foundations of Algorithms*", 4th Edition, Jones a Bartlett, Student Edition.
12. Reema Thareja, "*Data Structures using C*", Oxford Univ. Press

MCA Semester III, Paper-I (04 Credits)			
Core Course: MCA-3001 COMPUTER NETWORK			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
This course aims to provide students with a comprehensive understanding of computer networks, covering concepts such as network architecture, protocols, security, and troubleshooting. Students will gain practical skills in designing, configuring, and managing computer networks to support efficient data communication and collaboration.			

Block I	**Unit 1:** Introductory Concepts: History, Goals and Applications of Networks, Layered Network Architecture, Review of ISO-OSI Model **Unit 2:** Introduction to TCP/IP Model, Data Communication Techniques, Pulse Code Modulation (PCM) **Unit 3:** Multiplexing Techniques, Frequency Division, Time Division, Statistical Time Division Multiplexing. **Unit 4:** Physical Layer: Transmission Media: Wires, Cables, Radio Links, Satellite Link, Fiber Optic **Unit 5:** Error Detection and Correction: Single and Burst Error, Parity Check Codes, Cyclic Redundancy Code & Hamming Code	
Block II	**Unit 1:** Data Link Layer Protocols, Stop and Wait Protocols **Unit 2:** Noise free and Noisy channel, Performance, and efficiency, sliding Window Protocols **Unit 3:** Go Back and Selective Repeat ARQS, performance and efficiency.	
Block III	**Unit 1:** Medium access sub layer: Channel allocations, LAN protocols, ALOHA Protocols Pure ALOHA, slotted ALOHA **Unit 2:** Carrier Sense Multiple Access Protocols, CSMA with Collision Detection, Collision free Protocols **Unit 3:** IEEE Standards, FDDI, Data Link Layer elementary data link protocols, error handling **Unit 4:** High Level Data Link Control, DQDB. HDLC data link protocols, ISDN, Channel Structure, Asynchronous Transfer Mode ATM	
Block IV	**Unit 1:** Network and Transport Layer Protocols: General Principles, Virtual Circuits, and datagram's, Windows flow control, Packet Discarding, Traffic Shaping, Choke RSVP, Network Layer in ATM **Unit 2:** Internetworking using Bridge, Router and Gateways, Routing Algorithms: shortest path routing, Quality of Services, Primitives Connection Management: Addressing, Connection Establishment and Releases **Unit 3:** Flow Control and Buffering, Crash recovery, Element of TCP/IP protocol: User Data gram Protocol, (UDP/TCP) Layering. TCP/IP packet, IP addresses **Unit 4:** IPv6 Transport Layer: Design issues, connection management, TCP window Management, User Datagram Protocol, Transmission Control Protocol	
Block V	**Unit 1:** Application Layer: Network Security, DES, RSA algorithms, Domain Name System **Unit 2:** Simple Network Management Protocol, Electronic mail **Unit 3:** File Transfer Protocol, Hyper Text Transfer Protocol **Unit 4:** Cryptography and compression Techniques	

Suggested Readings:

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
5. Forouzan, "*Data Communication and Networking*", TMH

MCA Semester III, Paper-II (04 Credits)			
Core Course: MCA-3002 ARTIFICIAL INTELLIGENCE			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to provide students with a comprehensive understanding of Artificial Intelligence (A.I) concepts, methodologies, and applications. Through a combination of theoretical lectures, hands-on projects, and practical exercises, students will delve into the core principles of A.I, including machine learning, neural networks, and deep learning.

Block I	**Unit 1: INTRODUCTION:** Definitions, Basic Elements of Artificial Intelligence **Unit 2:** Artificial Intelligence application Areas, Intelligent Agents **Unit 3:** Structure of Intelligent Agents, natural language, Automated reasoning, visual perception	
Block II	**Unit 1: INTRODUCTION TO SEARCH:** search knowledge, Problem solving: Solving problems by searching: state space formulation, depth first and breadth first search. **Unit 2:** Iterative deepening production systems, search space control; depth-first, breadth-first search **Unit 3:** Heuristic Based Search: Heuristic search, Hill climbing, best-first search. **Unit 4:** branch and bound, Problem Reduction, Constraint Satisfaction End and Means-End Analysis	
Block III	**Unit 1: KNOWLEDGE REPRESENTATION AND REASONING:** Propositional logic **Unit 2:**Theory of first order logic, Inference in First order logic **Unit 3:** Forward & Backward chaining, Resolution **Unit 4:** Probabilistic reasoning, Utility theory, Hidden Markov Models (HMM)	
Block IV	**Unit 1: NATURAL LANGUAGE PROCESSING:** Introduction, Syntactic Processing, Semantic Processing, Pragmatic Processing **Unit 2:** Game Playing: Minimax, alpha-beta pruning Probabilistic reasoning systems. **Unit3:** Bayesian networks. Learning from observations: Inductive learning, learning decision trees. **Unit4:** Computational learning theory, Explanation based learning. Applications: Environmental Science, Robotics, Aerospace, Medical Science etc.	

Suggested Readings:

1. E. Rich and K. Knight, "*Artificial Intelligence*", Tata McGraw Hill.
2. E. Charnaik and D. McDermott, "*Introduction to Artificial Intelligence*", Addison Wesley Publishing Company.
3. Dan W. Patterson, "*Introduction to Artificial Intelligence and Expert Systems*", PHI.
4. W. F. Clofisin and C. S. Mellish, "*Programming in PROLOG*", Narosa Publishing Co.
5. Sanjiva Nath, "*Turbo PROLOG*", Galgotia Publications Pvt. Ltd.
6. K M Fu, "*Neural Networks in Computer Intelligence*", McGraw-Hill
7. Russel and Norvig, "*AI – A modern Approach*", Pearson Education

MCA Semester III, Paper-III (04 Credits)			
Core Course: MCA-3003 SOFTWARE ENGINEERING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to equip students with the essential knowledge, skills, and principles of software engineering to excel in designing, developing, and maintaining high-quality software systems. Through a combination of theoretical lectures, practical exercises, and hands-on projects, students will delve into the core concepts of software engineering, including requirements engineering, software design, coding, testing, and maintenance.

Block I	**Unit 1:** Introduction: Software Crisis, Software Processes & Characteristics, Software life cycle models, Waterfall, Prototype, Evolutionary and Spiral Models **Unit 2:** Overview of Quality Standards like ISO9001, SEI–CMM. Software Requirements analysis & specifications: Requirement engineering, requirement elicitation techniques like FAST, QFD & Use case approach **Unit 3:** Requirements analysis using DFD, Data dictionaries & ER Diagrams **Unit 4:** Requirements documentation, Nature of SRS, Characteristics & organization of SRS	
Block II	**Unit 1:** Software Project Planning Size Estimation like lines of Code & Function Count, Cost Estimation Models, Static single & Multivariable Models **Unit 2:** COCOMO, COCOMO-II, Putnam resource allocation model, Risk Management **Unit 3:** Software Design: Cohesion & Coupling, Classification of Cohesiveness & Coupling **Unit 4:** Function Oriented Design, Object Oriented Design, User Interface Design	
Block III	**Unit 1:** Software Metrics: Software measurements: What & Why, Token Count, Halstead Software Science Measures, Design Metrics, Data Structure Metrics **Unit 2:** Information Flow Metrics, Software Testing: Testing process, Design of test cases, functional testing: Boundary value analysis, Equivalence class testing, Decision table testing **Unit 3:** Cause effect graphing, Structural testing, Path Testing, Data flow and mutation testing, Unit Testing **Unit 4:** Structural testing, Path Testing, Data flow and mutation testing, Unit Testing, Integration and System Testing, Debugging, Alpha & Beta Testing, Regression Testing, Testing Tools & Standards.	
Block IV	**Unit 1:** Software Reliability: Importance, Hardware Reliability & Software Reliability, Failure and Faults **Unit 2:** Reliability Models, Basic Model, Logarithmic Poisson Model Calendar time Component **Unit 3:** Software Maintenance: Management of Maintenance, Maintenance Process, Maintenance Models **Unit 4:** Reverse Engineering, Software Re-engineering, Configuration Management, Documentation	

Suggested Readings:

1. K. K. Aggarwal & Yogesh Singh, "*Software Engineering*", New Age International, 2001.
2. R. S. Pressman, "*Software Engineering-A Practitioner's Approach*", 5th Ed., McGraw-Hill Int. Ed., 2001.
3. R. Fairley, "*Software Engineering Concepts*", Tata McGraw Hill, 1997.
4. P. Jalote, "*An Integrated approach to Software Engineering*", Narosa, 1991.
5. Stephen R. Schach, "*Classical & Object-Oriented Software Engineering*", IRWIN, 1996.
6. James Peter, W. Pedrycz, "*Software Engineering*", John Wiley & Sons., 1999
7. I. Sommerville, "*Software Engineering*", Addison Wesley, 1999

MCA Semester III, Paper-IV (04 Credits)			
Elective-1 Course: MCA-3004 DATA WAREHOUSING AND DATA MAINING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
This course aims to provide students with a comprehensive understanding of Data Warehousing and Data Mining concepts, techniques, and methodologies. Students will explore the process of designing, building, and managing data warehouses to facilitate efficient storage, retrieval, and analysis of large volumes of structured and unstructured data.			

Block I	**Unit 1:** Introduction: Data Warehouse Fundamentals: Introduction to Data Warehouse, OLTP Systems; Differences between OLTP Systems and Data Warehouse, **Unit 2:** Differences between OLTP Systems and Data Warehouse, Characteristics of Data Warehouse **Unit 3:** Functionality of Data Warehouse, Data Warehouse Architecture: Introductions **Unit 4:** Components of Data warehouse Architecture ,Advantages and Applications of Data Warehouse.	
Block II	**Unit 1:** Planning and Designing**:** Data Warehouse Planning and Requirements: Planning Data Warehouse and Key Issues **Unit 2:** Data Warehouse development Life Cycle, Dimensional Modeling: Data Warehouse Schemas; Star Schema **Unit 3:** Inside Dimensional Table, Inside Fact Table, Snowflake Schema	
Block III	**Unit 1:** Data Warehouse & OLAP: Introduction to OLAP, Characteristics of OLAP **Unit 2:**Steps in the OLAP Creation Process **Unit 3:** OLAP Architectures, Types of OLAP: MOLAP, ROLAP, HOLAP; Advantages of OLAP; Meta data	
Block IV	**Unit 1:** Scope of Data Mining, Predictive Modeling **Unit 2:** Architecture for Data Mining, Data Mining Tools	
Block V	**Unit 1:** Data Mining Versus Database Management System **Unit 2:** Data Mining Techniques:- Association rules **Unit 3:** Classification, Regression, Clustering	

Suggested Readings:

1. Alex Berson, Stephen J. Smith, "*Data Warehousing, Data mining & OLAP*", TMH
2. Mark Humphries, Michael W. Hawkins, Michelle C. Dy, "*Data Warehousing: Architecture and Implementation*", Pearson
3. I. Singh, "*Data Mining and Warehousing*", Khanna Publishing House

MCA Semester III, Paper-IV (04 Credits)			
Elective-1 Course: MCA-3005 CLOUD COMPUTING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

Cloud computing refers to the delivery of computing services such as storage, processing power, and software over the internet. It allows users to access data and applications remotely, reducing the need for physical infrastructure and providing scalability, flexibility, and cost-efficiency for businesses and individuals.

Block I	**Unit 1:** Cloud Computing Overview Origins of Cloud computing – Cloud components **Unit 2:** Essential characteristics – On-demand self-service, Broad network access, Location independent resource pooling ,Rapid elasticity , Measured service. **Unit 3:** Comparing cloud providers with traditional IT service providers, Roots of cloud computing.	
Block II	**Unit 1**: Cloud Insights Architectural influences – High-performance computing, Utility and Enterprise grid computing. **Unit 2:** Cloud scenarios – Benefits: scalability ,simplicity ,vendors ,security, Limitations – Sensitive information **Unit 3:** Application development- security level of third party - security benefits, Regularity issues: Government policies.	
Block III	**Unit 1:** Cloud Architecture - Layers and Models Layers in cloud architecture, Software as a Service (SaaS) **Unit 2:** Features of SaaS and benefits, Platform as a Service (PaaS), features of PaaS and benefits, Infrastructure as a Service (IaaS), features of IaaS and benefits **Unit 3:** Service providers, challenges, and risks in cloud adoption. Cloud deployment model: Public clouds – Private clouds – Community clouds **Unit 4:** Hybrid clouds - Advantages of Cloud computing.	
Block IV	**Unit 1:** Cloud Security- Security Patterns for Cloud Computing, Trusted Platform **Unit 2:** Geo-tagging, Cloud VM Platform Encryption **Unit 3:** Trusted Cloud Resource Pools ,Secure Cloud Interfaces, Cloud Resource Access Control **Unit4:** Cloud Data Breach Protection, Permanent Data Loss Protection.	
Block V	**Unit 1:** Application Development: Service creation environments to develop cloud-based applications. **Unit 2:** Development environments for service development. **Unit 3:** Amazon, Azure, Google App, Salesforce.com, IBM Cloud, Google Map Reduce, Yahoo Hadoop.	

Suggested Readings:

1. Anthony T. Velte, Toby J. Velte Robert Elsenpeter, "*Cloud computing a practical approach*", TATA McGraw- Hill, New Delhi, 2010
2. Michael Miller, "*Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online*" Que 2008
3. Judith Hurwitz, Robin Bloor, Marcia Kaufman, Fern Halper, "*Cloud Computing for Dummies*", Wiley Publishing, Inc,2010
4. Rajkumar Buyya, "*Cloud Computing (Principles and Paradigms)*", James Broberg, Andrzej Goscinski, John Wiley & Sons, Inc. 2011

MCA Semester III, Paper-V (04 Credits)		
Elective-2 Course: MCA-3006 BIG DATA		
Credit: 4	**CIA: 25** **ESE: 75**	**Max. Marks: 100**
A Big Data course offers a comprehensive exploration of the principles, technologies, and applications essential for managing and deriving insights from large and complex datasets. Students delve into the fundamental characteristics of Big Data, including volume, velocity, variety, veracity, and value, while learning about storage solutions like NoSQL databases and distributed file systems such as Hadoop, as well as processing frameworks like Apache Spark and Map Reduce.		
Block I	**Unit 1:** Introduction to Big Data, types of Digital Data, Characteristics of Data , Evolution of Big Data, Definition of Big Data, Challenges with Big Data **Unit 2:** 5Vs of Big Data, Business Intelligence vs. Big Data **Unit 3:** Big Data Analytics: Classification of analytics, Data Science, Terminologies in Big Data.	
Block II	**Unit 1:** Introduction to Hadoop: Features, Advantages, Versions, Overview of Hadoop Eco systems **Unit 2:** Hadoop distributions, Hadoop vs. SQL, RDBMS vs. Hadoop, Hadoop Components	
Block III	**Unit 1:** Hadoop Distributed File System: The Design of HDFS, HDFS Concepts, Command Line Interface **Unit 2:** Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives. **Unit 3:** Hadoop I/O: Compression, Serialization, Avro, and File-Based Data structures.	
Block IV	**Unit 1:** MapReduce: MapReduce Types and Formats, Map Reduce Features, Mapper, Reducer, Combiner. **Unit 2:** Partitioner, Searching, Sorting, Compression.	
Block V	**Unit 1:** Hadoop Eco systems: Pig : Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases **Unit 2**: Hive: Hive Shell, Hive Services, Hive Meta store, Comparison with Traditional Databases **Unit 3**:HiveQL, Tables, Querying Data and User Defined Functions. Big SQL : Introduction	

Suggested Readings:

1. Seema Acharya, Subhashini Chellappan, "*Big Data and Analytics*", Wiley Publication, 2015
2. Tom White, "*Hadoop: The Definitive Guide*" Third Edit on, O'reily Media, 2012.
3. Anand Rajaraman and Jeffrey David Ullman, "*Mining of Massive Datasets*", CUP, 2012.
4. Bill Franks, "*Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics*", John Wiley& sons, 2012.
5. Glenn J. Myatt, "*Making Sense of Data*", John Wiley & Sons, 2007

MCA Semester III, Paper-V (04 Credits)			
Elective-2 Course: MCA-3007 DIGITAL IMAGE PROCESSING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
This course aims to provide students with a solid foundation in Digital Image Processing (DIP) principles, techniques, and applications. Through theoretical lectures, practical demonstrations, and hands-on exercises, students will explore fundamental concepts such as image representation, enhancement, restoration, segmentation, and compression.			
Block I	**Unit 1:** Fundamentals: Need for DIP-Fundamental steps in DIP - Image Sampling and Quantization - Imaging geometry, discrete image mathematical characterization. **Unit 2:** Elements of visual perception-Image sensing and Acquisition **Unit 3:** Image Sampling and Quantization **Unit 4:** Imaging geometry, discrete image mathematical characterization		
Block II	**Unit1:** Two-dimensional Fourier Transform-Properties–Fast Fourier Transform **Unit 2:** Inverse FFT, Discrete cosine transform and KL transform. **Unit 3:** Discrete Short time Fourier Transform, Wavelet Transform-Discrete wavelet Transform-and its application in Compression		
Block III	**Unit 1:**Image Enhancement: Spatial Domain: Basic relationship between pixel Basic Gray level Transformations – Histogram Processing **Unit 2:** Smoothing spatial filters- Sharpening spatial filters. Frequency Domain: Smoothing frequency domain filters **Unit 3:** Sharpening frequency domain filters Homomorphic filtering		
Block IV	**Unit 1:** Image Restoration: Overview of Degradation models **Unit 2:** Unconstrained and constrained restorations-Inverse Filtering, **Unit 3**: Inverse Filtering, Wiener Filter		
Block V	**Unit 1:** Feature Extraction**:** Detection of discontinuities – Edge linking and Boundary detection. **Unit 2:** Thresholding-Edge based segmentation, Region based Segmentation, matching. **Unit 3:** Advanced optimal border and surface detection-Use of motion in segmentation **Unit 4:** Image Morphology, Boundary Descriptors-Regional descriptors.		
Block VI	**Unit 1:** Image Reconstruction from Projections**:** Need-Radon Transform-Back projection operator. **Unit 2:** Projection Theorem-Inverse Radon Transform		

Suggested Readings:

1. Rafael C. Gonzalez & Richard E. Woods, "*Digital Image Processing*", Pearson Education, 2/e, 2004.
2. Anil. K. Jain, "*Fundamentals of Digital Image Processing*", Pearson Education, 2003.

MCA Semester IV, Paper-I (04 Credits)		
Elective-3 Course: MCA-4001 SOFT COMPUTING		
Credit: 4	**CIA: 25**	**ESE: 75** **Max. Marks: 100**
This course aims to provide students with a comprehensive understanding of soft computing techniques such as neural networks, fuzzy logic, and genetic algorithms. Through theoretical concepts and practical applications, students will learn to analyze, design, and implement soft computing models to tackle complex real-world problems efficiently. By the end of the course, students will be proficient in utilizing soft computing methodologies for tasks including pattern recognition, data analysis, optimization, and decision-making across various domains.		
Block I	**Unit 1:** Neural Networks: History, overview of biological Neuro-system, Mathematical Models of Neurons **Unit 2:** ANN architecture, learning rules, Learning Paradigms-Supervised, Unsupervised and reinforcement Learning. **Unit 3:** ANN training Algorithms-perceptions, Training rules, Delta, Back Propagation Algorithm **Unit 4:** Multilayer Perceptron Model, Hopfield Networks, Associative Memories, Applications of Artificial Neural Networks	
Block II	**Unit 1:** Fuzzy Logic: Introduction to Fuzzy Logic, Classical and Fuzzy Sets: Overview of Classical Sets, **Unit 2:** Membership Function, Fuzzy rule generation. Operations on Fuzzy Sets: Compliment, Intersections, Unions, **Unit 3:** Unions, Combinations of Operations, Aggregation Operations **Unit 4:** Fuzzy Arithmetic: Fuzzy Numbers, Linguistic Variables, Arithmetic Operations on Intervals & Numbers, Lattice of Fuzzy Numbers, Fuzzy Equations.	
Block III	**Unit 1:** Fuzzy Logic: Classical Logic, Multivalued Logics, Fuzzy Propositions, Fuzzy Qualifiers. **Unit 2:** Linguistic Hedges. Uncertainty based Information: Information & Uncertainty. **Unit 3:** Non specificity of Fuzzy & Crisp Sets, Fuzziness of Fuzzy Sets. **Unit 4:** Introduction of Neuro-Fuzzy Systems: Architecture of Neuro Fuzzy Networks. Application of Fuzzy Logic: Medicine, Economics etc.	
Block IV	**Unit 1:** Genetic Algorithm: An Overview, GA in problem solving, Implementation of GA	

Suggested Readings:

1. Anderson J. A., "*An Introduction to Neural Networks*", PHI, 1999.
2. Hertz J. Krogh, R.G. Palmer, "*Introduction to the Theory of Neural Computation*", Addison-Wesley, California, 1991.
3. G. J. Klir & B. Yuan, "*Fuzzy Sets & Fuzzy Logic*", PHI, 1995.
4. Melanie Mitchell, "*An Introduction to Genetic Algorithm*", PHI, 1998.
5. "*Neural Networks - A Comprehensive Foundations*", Prentice-Hall International, New Jersey, 1999.
6. Freeman J. A. & D. M. Skapura, "*Neural Networks: Algorithms, Applications and Programming Techniques*", Addison Wesley, Reading, Mass, (1992).

MCA Semester IV, Paper-I (04 Credits)			
Elective-3 Course: MCA-4002 SOFTWARE QUALITY ENGINEERING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
The purpose of Software Quality Engineering for MCA students is to equip them with industry-standard methodologies, tools, and ethical principles for ensuring software quality. Through understanding quality standards, risk management, and process improvement, students are prepared to contribute effectively to high-quality software development projects in their future careers.			
Block I	**Unit 1:** Introduction Defining Software Quality, Software Quality Attributes and Specification, Cost of Quality, Defects, Faults **Unit 2:** Failures, Defect Rate and Reliability, Defect Prevention, Reduction, and Containment, **Unit 3:** Overview of Different Types of Software Review, Introduction to Measurement and Inspection Process, Documents and Metrics.		
Block II	**Unit 1:** Software Quality Metrics Product Quality Metrics: Defect Density, Customer Problems Metric **Unit 2:** Customer Satisfaction Metrics, Function Points, In-Process Quality Metrics: Defect Arrival Pattern, **Unit 3:** Phase-Based Defect Removal Pattern, Defect Removal Effectiveness, Metrics for Software Maintenance: Backlog Management Index, **Unit 4**: Fix Response Time, Fix Quality, Software Quality Indicators.		
Block III	**Unit 1:** Software Quality Management and Models Modeling Process **Unit 2:** Software Reliability Models: The Rayleigh Model, Exponential Distribution and Software Reliability Growth Models **Unit 3:** Software Reliability Allocation Models, Criteria for Model Evaluation, **Unit 4:** Software Quality Assessment Models: Hierarchical Model of Software Quality Assessment		
Block IV	**Unit 1:** Software Quality Assurance Quality Planning and Control, Quality Improvement Process **Unit 2:** Evolution of Software Quality Assurance (SQA), Major SQA Activities, Major SQA Issues **Unit 3:** Zero Defect Software, SQA Techniques, Statistical Quality Assurance **Unit 4:** Total Quality Management, Quality Standards and Processes.		
Block V	**Unit 1:** Software Verification, Validation & Testing: Verification and Validation, Evolutionary Nature of Verification and Validation, **Unit 2:** Impracticality of Testing all Data and Paths, Proof of Correctness, Software Testing, Functional, Structural and Error-Oriented Analysis & Testing **Unit 3:** Static and Dynamic Testing Tools, Characteristics of Modern Testing Tools.		

Suggested Readings:

1. Jeff Tian, "*Software Quality Engineering (SQE)*", Wiley-Inderscience, 2005; ISBN 0-471-71345-7.
2. Stephen H. Kan, "*Metrics and Models in Software Quality Engineering*", Addison Wesley (2002), ISBN: 0201729156

MCA Semester IV, Paper-II (04 Credits)			
Elective-4 Course: MCA-4003 NEURAL NETWORKS			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**
The objective of neural networks from MCA (Master of computer application) student is to comprehend the principles and architectures of artificial neural networks, enabling them to design , train, and deploy models for various tasks such as classification, regression, and pattern recognition, advancing their skills in machine learning and artificial intelligence.			

Block I	**Unit 1:** Fundamentals of ANN: Biological neurons, Feedforward Networks, Feedback Networks, Neural processing **Unit 2:** Supervised and unsupervised learning, Neural Network Learning Rules Hebbian Learning Rule **Unit 3:** Perceptron Learning Rule, Delta Learning Rule, Widrow-Hoff Learning Rule **Unit 4:** Correlation Learning Rule, Winner-Take-All Learning Rule, Outstare Learning Rule	
Block II	**Unit 1:** Classification Model, Features, Discriminant Functions **Unit 2:** Linear Machine and Minimum Distance Classification, Nonparametric Training Concept **Unit 3:** Single-Layer Continuous Perceptron Networks for Linearly Separable Classifications	
Block III	**Unit 1:** Linearly Non separable Pattern Classification, Delta Learning Rule for Multi-perceptron Layer, Generalized Delta Learning Rule, Feedforward Recall and Error Back- Propagation Training **Unit 2:** Feedforward Recall, Error Back-Propagation Training, Multilayer Feedforward Networks as Universal Approximators. **Unit 3:** Learning Factors Initial Weights, Cumulative Weight Adjustment versus, Incremental Updating, Steepness of the Activation Function, Learning Constant, Momentum Method, **Unit 4:** Network Architectures Versus Data Representation, Necessary Number of Hidden Neurons **Unit 5:** Classifying and Expert Layered Networks- Character Recognition Application, Expert Systems Applications	
Block IV	**Unit 1:** Single-Layer Feedback Networks: Basic Concepts of Dynamical Systems, Mathematical Foundations of Discrete-Time, Hopfield Networks, **Unit 2:** Mathematical Foundations of Gradient-Type Hopfield Networks **Unit 3:** Transient Response of Continuous-Time Networks, Relaxation Modelling in Single-Layer Feedback Networks, **Unit 4:** Example Solutions of Optimization Problems, Minimization of the Travelling Salesman Tour Length.	

Suggested Readings:

1. Jacek M. Zurada, "*Introduction to Artificial Neural Systems*", ISBN 0-3 14-93391-3, West Publishing Company.
2. Simon Haykin, "*Neural Networks - A Comprehensive Foundation*", 2nd Edition, ISBN 81-7808-300-0, Pearson Education (Singapore) Pte. Ltd.
3. G´erard Dreyfus, "*Neural Networks: Methodology and Applications*", ISBN-10 3-540-22980-9, Springer Verlag.
4. Kishan Mehrotra, Chilukuri K. Mohan, and Sanjay Ranka, "*Elements of Artificial Neural Networks*", ISBN 0- 262-13328-8

MCA Semester IV, Paper-II (04 Credits)			
Elective-4 Course: MCA-4004 INTERNET OF THINGS			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to provide students with a comprehensive understanding of the Internet of Things (IoT) ecosystem. Students will learn about IoT architectures, protocols, and technologies enabling the interconnection of physical devices. Practical skills will be developed through hands-on experience in designing, implementing, and managing IoT systems.

Block I	**Unit 1:** Internet of Things (IoT): Vision, Definition, Conceptual Framework **Unit 2:** Architectural view, technology behind IoT, Sources of the IoT, **Unit 3:** M2M Communication, IoT Examples. Design Principles for Connected Devices: IoT/M2M systems layers and design standardization. **Unit 4:**communication technologies, data enrichment and consolidation, ease of designing and affordability	
Block II	**Unit 1**: Hardware for IoT: Sensors, Digital sensors, actuators, radio frequency identification (RFID)technology **Unit 2:** Wireless sensor networks, participatory sensing technology. **Unit 3:** Embedded Platforms for IoT: Embedded computing basics, Overview of IoT supported Hardware platforms such as Arduino. **Unit 4:** Net Arduino, Raspberry pi, Beagle Bone, Intel Galileo boards and ARM cortex.	
Block III	**Unit 1:** Network & Communication aspects in IoT: Wireless Medium access issues **Unit 2:** MAC protocol survey, Survey routing protocols, **Unit 3:** Sensor deployment & Node discovery, Data aggregation & dissemination	
Block IV	**Unit 1:** Programming the Arduino: Arduino Platform Boards Anatomy, Arduino IDE **Unit 2:** Coding using emulator, using libraries, additions in Arduino, programming the Arduino for IoT.	
Block V	**Unit 1:** Challenges in IoT Design Challenges: Development Challenges, Security Challenges, Other challenges IoT Applications: Smart Metering, E-health, City Automation, **Unit 2:** Automotive Applications, home automation, smartcards **Unit 3:** Communicating data with H/W units, mobiles, tablets, Designing of smart streetlights in smart city.	

Suggested Readings:

1. Olivier Hersent, David Boswarthick, Omar Elloumi, "*The Internet of Things key applications and protocols*", willey
2. Jeeva Jose, "*Internet of Things*", Khanna Publishing House
3. Michael Miller, "*The Internet of Things*" Pearson
4. Raj Kamal, "*INTERNET OF THINGS*", McGraw-Hill,1st Edition, 2016
5. Arshdeep Bahga, Vijay Madisetti, "*Internet of Things (A hands on approach)*", 1st edition, VPI publications, 2014

MCA Semester IV, Paper-III (04 Credits)			
Elective-5 Course: MCA-4005 MACHINE LEARNING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

This course aims to provide students with a solid foundation in machine learning concepts, algorithms, and techniques. Students will learn to understand, implement, and evaluate supervised and unsupervised learning methods, including regression, classification, clustering, and dimensionality reduction. Practical skills will be developed through hands-on experience with popular machine learning libraries and frameworks.

Block I	**Unit 1: INTRODUCTION TO MACHINE LEARNING:** Introduction, Examples of various Learning Paradigms **Unit 2:** Perspectives and Issues, Version Spaces, Finite and Infinite Hypothesis Spaces **Unit 3**: PAC Learning, VC Dimension.	
Block II	**Unit 1: SUPERVISED LEARNING ALGORITHMS:** Learning a Class from Examples, Linear, Non-linear, Multi-class and multi-label classification. **Unit 2**: Decision Trees:ID3, Classification and Regression Trees (CART) **Unit 3**: Regression: Linear Regression, Multiple Linear Regression, Logistic Regression.	
Block III	**Unit 1: ENSEMBLELEARNING:** Ensemble Learning Model Combination Schemes **Unit 2:** Voting, Error-Correcting Output Codes **Unit 3:** Bagging: Random Forest Trees, Boosting: Adaboost, Stacking	
Block IV	**Unit 1: UNSUPERVISED LEARNING:** Introduction to clustering, Hierarchical: AGNES, DIANA **Unit 2:** Partitional: K-means clustering, K-Mode Clustering, Self-Organizing Map, Expectation Maximization, Gaussian Mixture Models **Unit 3:** Principal Component Analysis (PCA), Locally Linear Embedding (LLE), Factor Analysis	

Suggested Readings:

1. Pradhan M., Kumar U. D., "*Machine Learning Using Python*", Wiley, 2019
2. Anuradha Srinivasaraghavan, Vincy Joseph, "*Machine Learning*", Wiley, 2019.
3. Saikat Dutt, S. Chandramouli, A. K. Das, "*Machine Learning*", Pearson, 2019.
4. Alex Smola and S.V.N. Vishwanathan, "*Introduction to Machine Learning*", Cambridge University Press, 2008.
5. Peter Harrington, "*Machine Learning in Action*", Manning Publications, 2012.
6. M. Mohammed, M. Badruddin Khan, E. Bashier M. Bashier, "*Machine Learning – Algorithms and Applications*", CRC Press, 2017.

MCA Semester IV, Paper-III (04 Credits)			
Elective-5 Course: MCA-4006 QUANTUM COMPUTING			
Credit: 4	**CIA: 25**	**ESE: 75**	**Max. Marks: 100**

The objective quantum computing for an MCA (Master of Computer applications) student is to grasp the fundamentals of quantum mechanics and quantum computations , enabling them to develop and apply quantum algorithms for solving computationally intensive problems efficiently, exploring the potential of quantum technologies for advanced computing solutions.

Block I	**Unit 1:** Introduction to Quantum Computation: Quantum bits, **Unit 2:** Bloch sphere representation of a qubit, multiple qubits.	
Block II	**Unit 1:** Background Mathematics and Physics: Hilber space, **Unit 2:** Probabilities and measurements, entanglement, density operators and correlation **Unit 3:** Basics of quantum mechanics **Unit 4:** Measurements in bases other than computational basis.	
Block III	**Unit 1:** Quantum Circuits: single qubit gates, **Unit 2:** Multiple qubit gates **Unit 3:** Design of quantum circuits	
Block IV	**Unit 1:** Quantum Information and Cryptography: Comparison between classical and quantum information theory **Unit 2:** Bell states, Quantum teleportation. **Unit 3:** Quantum Cryptography **Unit 4:** No cloning theorem	
Block V	**Unit 1:** Quantum Algorithms: Classical computation on quantum computers. Relationship between quantum and classical complexity classes **Unit 2:** Deutsch's algorithm, Deutsch's-Jozsa algorithm, **Unit 3:** Shor factorization, Grover search, Noise, and error correction: Graph states and codes **Unit 4:** Quantum error correction, fault-tolerant computation.	

Suggested Readings:

1. Nielsen M. A., "*Quantum Computation and Quantum Information*", Cambridge University Press.2002
2. Benenti G., Casati G., and Strini G., "*Principles of Quantum Computation, and Information*", *Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics*, World Scientific.2004
3. Pittenger A. O., "*An Introduction to Quantum Computing Algorithms*", 2000