Post Graduate Department of Computer Sciences,
University of Kashmir,
Srinagar-190006

Syllabus of One/Two Year MCA Programme
Effective from Year/Batch 2025

Eligibility Criteria

- **For Two-Year MCA: -** "Any Graduate with at least 12 credits in Computer Science / applications under CBCS/NEP 2020 Scheme

  OR

  B.Sc. with Mathematics (at 10+2 level OR in graduation) or B.Tech/BE"

- **For One-Year MCA:** "Any 4-Year graduate with at least 20 credits in Computer Science/applications or any other computing field".

Programme Learning Outcomes (PLOs)

1. **Knowledge and Understanding:**
   Evaluate core and advanced concepts in computing such as algorithms, system design, artificial intelligence, and data-driven systems, with the ability to interpret emerging trends and technologies.
2. **Technical Skills:**
   Design, implement, and evaluate software systems using modern computing tools, programming languages, and best practices in system development and deployment.
3. **Application of Knowledge and Skills:**
   Integrate computational knowledge with domain-specific problems to build and deploy practical solutions across sectors such as healthcare, environment, education, and business.
4. **Communication Skills:**
   Demonstrate the ability to effectively articulate technical ideas, research findings, and project outcomes in written, oral, and visual formats for diverse stakeholders.
5. **Critical Thinking:**
   Analyze complex technical problems, synthesize information from multiple sources, and apply logical reasoning to develop, test, and refine effective solutions.
6. **Ethics:**
   Apply ethical principles, legal guidelines, and professional standards in the design, implementation, and evaluation of computing technologies.
7. **Life-long Learning:**
   Pursue continuous professional growth and knowledge enhancement through research, certifications, conferences, and scholarly engagement.
8. **Creativity:**
   Design innovative computing models and user-centric solutions by applying creativity in algorithm development, interface design, and system architecture.
9. **Research Aptitude:**
   Formulate research problems, conduct literature reviews, design experimental setups, and evaluate results using appropriate scientific methods and tools.
10. **Problem Solving:**
    Identify real-world challenges, define computational problems clearly, and develop optimized, scalable solutions using algorithmic thinking and systematic evaluation.

Two Year MCA Syllabus Structure (CW+R)

Credit Level	Semester	Course Type	Course Code with Name	Course Level	Credits	Total Credits	Max. Marks			Credit Distribution	Contact Hours
							Continuous Assessment	End Semester	Total	L: T: P	
6.0	Sem - I	Core	MMCACJP125: Java Programming	400	4	22	28	72	100	4:0:0	60
		Core	MMCACML125: Machine Learning	400	4		28	72	100	4:0:0	60
		DCE-I	MMCADAD125: Advanced Data Structures MMCADCG125: Computer Graphics MMCADMI125: Management Information System MMCADSE125: Software Engineering	400	4		28	72	100	4:0:0	60
		DCE-II	MMCADDS125: Advanced Database Systems MMCADAI125: Artificial Intelligence MMCADBC125: Block Chain Technologies MMCADCS125: Cyber Security & Digital Forensics	400	4		28	72	100	4:0:0	60
		Core	MMCACRM125: Research Methodology	400	2		14	36	50	2:0:0	30
		Lab	MMCALJP125: Java Programming Lab	400	2		14	36	50	0:0:2	60
		Lab	MMCALML125: Machine Learning Lab	400	2		14	36	50	0:0:2	60
	Sem - II	Core	MMCACDA225: Design and Analysis of Algorithms	400	4	22	28	72	100	4:0:0	60
		Core	MMCACMA225: Mobile Application Development	400	4		28	72	100	4:0:0	60
		DCE-III	MMCADAO225: Advanced Operating Systems MMCADDI225: Digital Image Processing MMCADDS225: Decision Support Systems MMCADCN225: Cryptography & Network Security	400	4		28	72	100	4:0:0	60
		DCE-IV	MMCADAC225: Advanced Computer Networks MMCADCC225: Cloud Computing MMCADLP225: Linux Programming MMCADTC225: Theory of Computation	400	4		28	72	100	4:0:0	60
		Core	MMCACRP225: Research and Publication Ethics	400	4		28	72	100	4:0:0	60
		Lab	MMCALMA225: Mobile Application Development Lab	400	2		14	36	50	0:0:2	60
Total (First Year)					44	44	308	792	1100	38:0:6	750 Hrs
6.5	Sem - III	Core	MMCACDS325: Data Science with Python	500	4	22	28	72	100	4:0:0	60
		Core	MMCACWP325: Web Programming	500	4		28	72	100	4:0:0	60
		DCE-V	MMCADQC325: Quantum Computing MMCADEH325: Ethical Hacking MMCADCV325: Computer Vision MMCADER325: Enterprise Resource Planning	500	4		28	72	100	4:0:0	60
		DCE-VI	MMCADNL325: Natural Language Processing MMCADSQ325: Software Quality Assurance MMCADDL325: Deep Learning MMCADIT325: Internet of Things	500	4		28	72	100	4:0:0	60
		Core	MMCACSP325: Software Project Management	500	2		14	36	50	2:0:0	30
		Lab	MMCALDS325: Data Science with Python Lab	500	2		14	36	50	0:0:2	60
		Lab	MMCALWP325: Web Programming Lab	500	2		14	36	50	0:0:2	60
	Sem - IV	Project	MMCAPPI425: Problem Identification & Analysis	500	6	20	42	108	150	6:0:0	90
		Project	MMCAPDI425: Dissertation	500	6		42	108	150	6:0:0	90
		Project	MMCAPSD425: Software Development	500	4		28	72	100	0:0:4	120
		Project	MMCAPRC425: Research Component	500	4		28	72	100	0:0:4	120
Total (Second Year)					42	42	294	756	1050	30:0:12	810 Hrs
TOTAL CREDITS (AGGREGATE OF 4-SEMESTERS)					*86*	*86*	*602*	*1548*	*2150*	*68:0:22*	*156 0 Hrs*

SEMESTER - I

To be effective from year-2025

COURSE TITLE:JAVA PROGRAMMING

Course Code: MMCACJP125						**Examination Scheme**	
Total number of Lecture Hours: 60						**External**	**72**
						Internal	**28**
Lecture (L):	**4**	**Practicals(P):**	**-**	**Tutorial (T):**	**-**	**Total Credits**	

Course Objectives

- To understand the fundamental principles of Java programming language, including its syntax, semantics, and basic constructs.
- To explore object-oriented programming concepts such as classes, inheritance, polymorphism, and interfaces in the context of Java.
- To develop proficiency in handling exceptions and errors using Java's exception handling mechanisms.
- To gain practical experience in utilizing Java's standard library classes and packages for tasks like I/O operations, string manipulation, and multithreading.
- To learn to create graphical user interfaces (GUIs) in Java, employing event-driven programming paradigms and integrating various GUI elements.
- To acquire skills in network programming with Java, including socket programming for communication between distributed systems and applications.

Course Content	TEACHING HOURS
UNIT 1: Introduction to Java Programming	**15- Hrs**
Introduction to Java Language: Creation of Java. How Java changed the Internet. Features of Java Language. Evolution of Java. Comparison with other languages like C++.Java Virtual Machine (JVM) and Byte-code. Java Language Overview: Lexical issues – Whitespace, Identifiers, Keywords, Literals, Separators, and Comments. Installing JDK.PATH variable. Java program – Structure, Compilation and Execution. Java Class libraries (System Class).main() method. **Data types, Variables and Arrays:** Primitive Data-types and Typed-Literals. Variables – Declaration, Initialization, Scope and Lifetime. Arrays – Single and Multidimensional. Type Conversion and Expression Promotion. **Operators, Expressions and Control statements:** Arithmetic, Bitwise, Relational, Logical, Assignment. Precedence and Associativity. Selection, Iteration and Jump Statements.	
UNIT 2: Object-Oriented Programming in Java	**15- Hrs**
Class Fundamentals: Class Structure (Variable and Method declaration).Modifiers (Access Modifiers and Other Modifiers).Components of Class, Variable and Method declaration. Constructor and finalize(). Garbage Collection. Passing parameters to methods. Variable hiding. Method overloading. Constructor overloading and chaining. Use of this keyword. Code blocks - Static and non-static. **Inheritance:** Mechanism. Role of Access Modifiers. Method Overriding and Shadowing. Use of super keyword. Polymorphism - Early and Late binding. Abstract Class and Interface. Components of Interface declaration. Implementing Interfaces. **Exception Handling:** Mechanism - Exception-Object, Throwing an Exception, and Exception Handler. Catch or Specify policy. Types of Exception - Checked vs Unchecked, Built-in vs Userdefined. Catching an Exception - try-catch-finally.	

Specifying an Exception - throws. Manually throwing an Exception - throw. Custom Exceptions. Chained Exceptions.	
UNIT 3: Advanced Java Concepts	**15- Hrs**
Packages: Creating and Importing Packages. CLASSPATH variable. static import. **Strings:** Mutable and Immutable Strings. Creating Strings. Operations on Strings. **Threads:** Creating Threads in Java. Java Thread Lifecycle. Multithreading in Java: Synchronization and Inter- process communication (IPC) in Threads. **I/O Streams:** Byte, Character, Buffered, Data, and Object Streams. Standard Streams. File I/O Basics, Reading and Writing to Files. Serializing Objects.	
UNIT 4: Java GUI Programming and Networking	**15-Hrs**
Event-Driven Programming: Java 1.1 Event Delegation Model – Source object, Event object and Listener object. Methods associated with Source, Event and Listener objects. Low-level vs Semantic events. Adapter classes, Inner classes, and Anonymous Inner classes. Adding GUI elements to Applet. **Networking Classes and Interfaces:** TCP/IP Server Sockets in Java. Developing simple networking applications in Java like File transfer, Chatting, etc.	

Textbooks

1. H. Schildt, Java: The Complete Reference, 13th Edition, Tata McGraw Hill, 2023.

Reference Books

1. E. Balagurusamy, Programming with Java: A Primer, 7th Edition, Tata Mcgraw Hill, 2023.
2. H.M. Dietel and P.J. Dietel, Java: How to Program, 11th Edition, Pearson Education, 2017.
3. K. Sierra and B. Bates, Head First Java (Java 5), 2nd Edition, O'Reilly, 2003.
4. C.S. Horstmann and G. Cornell, Java 2 Vol-1 Fundamentals, 7th Indian Reprint, Pearson Education, 2006.

COURSE LEARNING OUTCOMES (CLO):

CLO1: Understanding of the foundational concepts of Java programming, including data types, control structures, program flow, and compilation/execution of Java applications.

CLO2: Apply object-oriented programming principles in Java using classes, inheritance, polymorphism, interfaces, and exception handling for robust application development.

CLO3: Demonstrate the use of advanced Java features such as multithreading, string manipulation, package management, and file I/O operations.

CLO4: Design and implement event-driven GUI applications and basic networking solutions using Java APIs and socket programming.

CLOs	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	Average (CLO)
MMCACJP125.1	3	2	1	1	2	1	1	0	2	2	1.5
MMCACJP125.2	2	3	2	1	3	1	1	2	1	3	1.9
MMCACJP125.3	2	3	3	1	3	1	2	2	1	3	2.1
MMCACJP125.4	2	3	3	2	3	1	2	3	1	3	2.3
Average(PLO)	2.25	2.75	2.25	1.25	2.75	1	1.5	1.75	1.25	2.75	1.95

To be effective from year-2025

COURSE TITLE: Machine Learning

Course Code: MMCACML125						Examination Scheme	T	P
Total number of Lecture Hours: 60						External	72	-
						Internal	28	-
Lecture (L):	4	Practical (P):	-	Tutorial (T):	-	Total Credits		4

Course Objectives

- To introduce the fundamental concepts, techniques, and applications of machine learning and provide insight into its challenges and testing methods.
- To equip students with the ability to build, evaluate, and optimize basic machine learning models including regression and classification models.
- To develop proficiency in various clustering techniques and feature engineering for unsupervised learning scenarios.
- To enable students to apply advanced classification methods such as Bayesian learning and Support Vector Machines for solving complex real-world problems.

Course Content	TEACHING HOURS
UNIT 1: Introduction to Machine Learning and Data Preprocessing	**15 Hrs.**
Machine Learning, Applications, Types of Learning, Main Challenges of Machine Learning, Testing and Validating, designing a learning System, Inductive Bias and Hypothesis, Hypothesis Evaluation, Feature extraction, Types of feature selection, Feature Handling, Normalization, Missing data, Dimensionality Reduction: Principal Component Analysis	
UNIT 2: Regression, Classification, and Clustering Basics	**15 Hrs.**
Linear Regression, Logistic Regression, Decision Tree Representation, The Basic Decision Tree Learning Algorithm, Hypothesis Space Search in Decision Tree Learning, Inductive Bias in Decision Tree Learning, Issues in Decision Tree Learning, Clustering Algorithms: Euclidean and Mahalanobis Distance, K-means algorithm	
UNIT 3: Advanced Clustering Techniques and Instance-Based Learning	**15Hrs.**
Cluster validity index, Compactness Cluster measure, Distinctness Cluster Measure, Fuzzy C-means, Hierarchical Clustering, Density based spatial clustering of applications with noise (DBSCAN), Spectral clustering, k-medoids clustering, Kohenon Self Organizing Net, K- Nearest Neighbour and effect of various Distance measures	
UNIT 4: Probabilistic Learning and Support Vector Machines	**15 Hrs.**
Bayesian Learning: (Bayes Theorem and Concept Learning, Maximum Likelihood and Least- Squared Error Hypothesis, Naïve Bayes Classifier, Bayesian Belief Networks). Support Vector Machine: Linear Support Vector Machine, Optimal Hyperplane, Kernel functions, Solving Non-Linear Classification problems with Linear Classifier, Multiclass Support Vector Machines, Applications of Support Vector	

Machines	

Textbooks

1. Machine Learning by Tom M. Mitchel, McGraw-Hill publication

Reference Books

1. Pattern Classification by Duda and Hart. John Wiley publication
2. The Elements of Statistical Learning: Data Mining, Inference, and Prediction by Trevor Hastie, Robert Tibshirani, Jerome Friedman, Springer.
3. Learning From Data, Yaser S. Abu-Mostafa, Hsuan-Tien Lin, Malik Magdon-Ismail, AML Book.
4. Introduction to Machine Learning by EthemAlpaydin, The MIT Press.
5. Machine Learning: An Algorithmic Perspective by Stephen Marsland, Chapman and Hall/CRC.

COURSE LEARNING OUTCOMES (CO):

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

CLO1: Understand the types, challenges, and foundations of machine learning, and apply preprocessing techniques such as feature selection, normalization, and dimensionality reduction.
CLO2: Implement and analyze basic supervised and unsupervised machine learning algorithms including regression, decision trees, and k-means clustering.
CLO3: Apply and evaluate advanced clustering techniques and instance-based learning models for complex pattern discovery.
CLO4: Implement and compare probabilistic models and support vector machines for classification and prediction in various application domains.

CLO-PLO Matrix for the Course

Unit-Wise CLOs	PLOs										
	1	2	3	4	5	6	7	8	9	10	Average (CLO)
MMCACML125.1	3	2	2	2	2	1	2	2	2	2	**2.0**
MMCACML125.2	3	3	3	2	3	1	2	2	2	3	**2.4**
MMCACML125.3	3	3	3	2	3	1	2	2	2	3	**2.4**
MMCACML125.4	3	3	3	2	3	1	2	2	2	3	**2.4**
Average (PLO)	**3.0**	**2.75**	**2.75**	**2.0**	**2.75**	**1.0**	**2.0**	**2.0**	**2.0**	**2.75**	**2.3**

DCE-I

To be effective from year-2025

COURSE TITLE: Advanced Data Structures									
Course Code: MMCADAD125							**Examination Scheme**	**T**	**P**
Total number of Lecture Hours: 60							**External**	**72**	**-**
							Internal	**28**	**-**
Lecture (L):	**4**	**Practical(P):**	**0**	**Tutorial (T):**	**0**		**Total Credits**	**4**	

Course Objectives

- To understand and implement fundamental linear data structures and algorithms, including arrays, searching, sorting, matrices, and linked lists.
- To apply stack and queue data structures to solve computational problems using array and linked list implementations.
- To analyze and implement tree and graph structures along with their traversal techniques and real-world applications.
- To explore advanced data structures and algorithms including hashing, heaps, and file organizations for optimized data processing.

Course Content	TEACHING HOURS
Unit I: Linear Data Structures	**15 Hrs.**
Data types/objects/structures, Data structures and its types, Representation and implementation. Linear Data Structures: Array representation, operations, applications and limitations of linear arrays, Searching Techniques- Linear Search, Binary Search Sorting Techniques- Selection, Insertion sort, Bubble sort, Quick Sort, Merge Sort Two dimensional arrays, matrices, common operations of matrices, special matrices, Array representation of Sparse matrices. Linked Lists: Representation, Types and operations on Linked List.	
Unit II: Stack and Queues	**15 Hrs.**
Stack- Representation of stack in memory, Operations on Stacks, Implementation of Stack using arrays and linked list, Multiple Stacks: Representing two stacks and more than two stacks, Applications of stacks: Parenthesis Checker, Infix to postfix procedure, evaluating expressions in postfix notation, Implementation of recursion using stack. Queues- Representation of Queue in Memory, Operations on Queue, Implementation of Queue using arrays and linked list, Circular Queue and its operations, Representation and implementation, Multiple Queues, Deque, Priority Queue, Heap Representation of a Priority Queue, Applications of Queues.	
Unit III: Tree and Graph Data Structures	**15 Hrs.**
Trees, Definitions, terminologies and properties, Binary tree representation, traversals and applications, Threaded binary trees, Binary Search Trees, AVL Trees, M-way Search Trees, B-trees, B+ trees. Graphs, Terminology, Graph representations, Traversal Techniques, Operations on Graphs, Applications of Graphs	
Unit IV: Advanced Data Structures and Algorithms	**15 Hrs.**

To be effective from year-2025

Minimum spanning trees, Shortest Path Algorithms in Graphs, Eulerian Tour, Hamiltonian Tour Hashing: Direct Address Tables, Hash Table, Different Hash functions, resolving collisions, rehashing, Heap Structures, Binomial Heaps, Leftist Heaps. File Organizations: Sequential File Organization, Relative File Organization, Indexed Sequential File Organization, Multiple Key File Organizations: Inverted File and Multi-List Organizations	

Textbooks

1. Langsam, Augenstein, Tenenbaum, "Data Structures Using C and C++", 2nd Edition, 2015

Reference Books

1. Ellis Horowitz, Sartaj Sahni, Susan Anderson Freed, "Fundamentals of Data Structures In C", 2nd Edition, 2018
2. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", 3rd Edition, 2007.
3. Aho Alfred V., Hopcroft John E., UIlman Jeffrey D, "Data Structures and Algorithms", 2017
4. R. S. Salaria, "Data Structures and Algorithms Using C++", 2018
5. Varsha H Patil, "Data Structures using C++", 2012
6. E.Balagurusamy, "Object Oriented Programming with C++" , 8th Edition, 2020

COURSE LEARNING OUTCOMES (CLO):

CLO1: Understand and implement fundamental linear data structures and algorithms, including arrays, searching, sorting, matrices, and linked lists.
CLO2: Apply stack and queue data structures to solve computational problems using array and linked list implementations.
CLO3: Analyze and implement tree and graph structures along with their traversal techniques and real-world applications.
CLO4: Explore advanced data structures and algorithms including hashing, heaps, and file organizations for optimized data processing.

CLO-PLO Matrix for the Course

	PLOs										
Unit-Wise CLOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Average (CLO)**
MMCADAD125.1	3	3	2	1	3	0	2	2	1	3	**2.0**
MMCADAD125.2	3	3	2	1	3	0	2	2	1	3	**2.0**
MMCADAD125.3	3	3	3	1	3	0	2	3	2	3	**2.3**
MMCADAD125.4	3	3	3	1	3	1	2	3	2	3	**2.4**
Average (PLO)	**3.0**	**3.0**	**2.5**	**1.0**	**3.0**	**0.25**	**2.0**	**2.5**	**1.5**	**3**	**2.2**

To be effective from year-2025

COURSE TITLE: Computer Graphics									
Course Code: MMCADCG125						**Examination Scheme**	**T**	**P**	
Total number of Lecture Hours: 60						**External**	**72**	-	
						Internal	**28**	-	
Lecture (L):	**4**	**Practical(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**		

Course Objectives:

- To understand key concepts, graphic display devices, and 2D/3D transformations.
- To develop skills in line and circle drawing, clipping, filling, and hidden surface removal.
- To apply mathematical techniques like splines and Bezier methods for complex graphical models.
- To explore multimedia concepts, file formats, storage solutions, and introductory AR/VR technologies.

Course Content	**TEACHING HOURS**
UNIT 1:	**15Hrs**
Introduction to Computer Graphics, Applications of Computer Graphics, Graphic Display Devices: Refresh Cathode Ray Tubes, Raster-scan Displays, Random-Scan displays, Color CRT Monitors, Concept of Double Buffering, Lookup tables. 2-D Graphics: Cartesian and Homogeneous Coordinate Systems, Line drawing algorithms (Bressenham's and DDA), Circle and Ellipse Drawing Algorithms.	
UNIT 2:	**15Hrs**
2-Dimensional Transformations, Concepts of Window & Viewport, Window to Viewport Transformations, Normalization transformation (3L) Composite Transformations: General pivot point rotation, General fixed point scaling, reflection w.r.t line $y=x$, reflection w.r.t line y=x (4L) Transformation between coordinate systems, affine transformations, Raster methods for transformations (3L)	
UNIT 3:	**15Hrs**
Filling techniques: Boundary and Flood-fill algorithms (2L) Clipping, Line Clipping Algorithms (Cohen-Sutherland Algorithm), 3-D Graphics, Projections: perspective and parallel projection transformations. (5L) 3-Dimensional Transformations, Hidden Surface Removal Techniques, Z-Buffer Algorithm, Back Face Detection (3L)	
UNIT 4:	**15Hrs**
Curves and Surfaces: Spline specification, Interpolated& Approximated Splines. spline representation, cubic spline interpolation methods, Bezier Splines, Bezier Curves, Cubic Bezier Curves, Bezier Surfaces. (3LIntroduction to multimedia elements: Images (BMP, PCX), sound (WAV, MP3) Multimedia storage formats: CDs and DVDs). Introduction to virtual reality (VR) and augmented reality (AR) technologies.	

Textbooks

1. Hearn and Baker, "Computer Graphics with OpenGL": 4th Edition (2022), Donald Hearn, M. Pauline Baker, Warren Carithers, Pearson
2. Ze-Nian Li and Mark S. Drew, "Fundamentals of Multimedia": 3rd Edition (2021), Springer.

To be effective from year-2025

3. **W.M. Newman and Sproull, "Principles of Interactive Computer Graphics"** McGraw-Hill Education; 3rd Edition, 2023.

Reference Books

1 **Steven Harrington, "Computer Graphics: A Programming Approach"** McGraw-Hill Education; 2nd Edition, 2021.
2 **Plastock and Kelley, "Schaum's Outline of Theory and Problems of Computer Graphics"** McGraw-Hill Education; 2nd Edition, 2022.
3 **David F. Rogers and J. Alan Adams, "Procedural Elements of Computer Graphics"** McGraw-Hill Education; 3rd Edition, 2021.
4 **David F. Rogers and J. Alan Adams, "Mathematical Elements of Computer Graphics"** McGraw-Hill Education; 3rd Edition, 2022.
5 **James D. Foley, Andries van Dam, et al., "Computer Graphics: Principles and Practice"** Pearson; 4th Edition, 2023.
6 **Sinha and Udai, "Computer Graphics"** Tata McGraw-Hill Education; 2nd Edition, 2022

COURSE LEARNING OUTCOMES (CLO):

CLO1: Understand basic computer graphics concepts, display devices, and 2-D drawing algorithms.
CLO2: Apply 2-D transformations, window-to-viewport mapping, and coordinate conversions.
CLO3: Implement filling, clipping algorithms, 3-D projections, and hidden surface removal.
CLO4: Analyze spline curves, multimedia basics, and introduction to VR/AR technologies.

LEVEL OF CO-PO MAPPING TABLE

	PLO										
UNIT-WISE CLOs	1	2	3	4	5	6	7	8	9	10	**Avg (CLO)**
MMCADCG125.1	3	2	0	0	0	0	0	0	0	0	**2.5**
MMCADCG125.2	0	3	3	2	0	0	0	0	0	0	**2.67**
MMCADCG125.3	0	0	0	0	3	3	0	0	0	0	**3.0**
MMCADCG125.4	0	0	0	0	0	0	3	3	0	0	**3.0**
Avg (PLO)	**1.5**	**1.67**	**1.5**	**1.0**	**1.5**	**1.5**	**1.5**	**1.5**	**0**	**0**	**2.79**

COURSE TITLE: Management Information System

Course Code: MMCADMI125						**Examination Scheme**	**T**	**P**	
Total number of Lecture Hours: 60						**External**	**72**	**-**	
						Internal	**28**	**-**	
Lecture (L):	**4**	**Practical(P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**4**		

Course Objectives

- To understand the structure of organizations and the role of various information systems (MIS, DSS, GDSS).
- To analyze system requirements using structured system analysis tools and methods.
- To Explore enterprise systems like ERP, SCM, and CRM, and their role in strategic IT decisions.
- To ethical Evaluate the ethical, security, and social issues surrounding the use of information systems.

Course Content	TEACHING HOURS
UNIT 1: Introduction to Organizations and Information Systems	**15 Hrs.**
Organization and Information Systems, The Organization: Structure, Managers and activities – Data ,information and its attributes – The level of people and their information needs - Types of Decisions and information - Information System, - Management Information System (MIS) –Decision Support System (DSS) and Group Decision Support System (GDSS).	
UNIT 2: System Analysis and Development	**15 Hrs.**
Need for System Analysis - Stages in System Analysis - Structured SAD and tools like DFD, Context level Diagram, Decision Table and Structured Diagram. System Development Models: Waterfall, Prototype, Spiral, –Roles and responsibilities of System Analyst, Database Administrator and Database Designer.	
UNIT 3: Enterprise Systems and IT Decision-Making	**15 Hrs.**
Enterprise Resources Planning (ERP): Features, selection criteria, merits, issues and challenges in Implementation - Supply Chain Management (SCM): Features, Modules in SCM – Customer Relationship Management (CRM): Phases. Knowledge Management and e-governance, Nature of IT decisions- Strategic decision.	
UNIT 4: Security, Ethics, and Social Challenges in Information Systems	**15 Hrs.**
Security and Ethical Challenges, Ethical responsibilities of Business Professionals – Business, technology, Computer crime – Hacking, cyber theft, unauthorized use at work. Issues and internet privacy. Challenges – working condition, health and social issues, Ergonomics and cyber terrorism.	

Textbooks

1. 1. "Management Information Systems", Kenneth J Laudon, Jane P. Laudon, Pearson/PHI,10/e, 2007
2. "Management Information Systems", W. S. Jawadekar, Tata McGraw Hill Edition, 3/e, 2004

Reference Books: -

1. Turban, Efraim, Ephraim McLean, and James Wetherbe. 2007. Information Technology for Management: Transforming Organizations in the Digital Economy. New York, John Wiley & Sons.

COURSE LEARNING OUTCOMES (CLO):

CLO1: Describe the organizational structure and classify different types of information systems based on managerial needs.
CLO2: Apply system analysis and design techniques including DFDs, context diagrams, and decision tables.
CLO3: Evaluate and compare enterprise systems such as ERP, SCM, and CRM and understand IT-based strategic decisions.
CLO4: Analyze ethical, legal, and security issues in the management and use of information systems.

LEVEL OF CO-PO MAPPING TABLE

	PLOs										
Unit wise CLOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Avg (CLO)**
MMCADMI125.1	3	2	2	2	2	2	1	2	1	2	1.9
MMCADMI125.2	2	3	3	3	3	1	1	1	2	2	2.1
MMCADMI125.3	2	2	2	2	2	3	2	2	2	3	2.2
MMCADMI125.4	2	2	2	2	2	3	2	3	2	3	2.3
Avg (PLO)	**2.25**	**2.25**	**2.25**	**2.25**	**2.25**	**2.25**	**1.5**	**2.0**	**1.75**	**2.5**	**2.1**

To be effective from year-2025

COURSE TITLE: Software Engineering								
Course Code: MMCADSE125						**Examination Scheme**	**T**	**P**
Total number of Lecture Hours: 60						**External**	**72**	**-**
						Internal	**28**	**-**
Lecture (L):	**4**	**Practical (P):**	**-**	**Tutorial (T):**	**-**	**Total Credits**		**4**

Course Objectives

- To Gain knowledge of the nature, goals, and challenges of software engineering and its historical context.
- To Learn and utilize various software development models, including Waterfall, Agile, and Spiral.
- To Analyze software processes using measures, metrics, and models like CMMI and COCOMO.
- To Develop skills in eliciting, analyzing, modeling, and validating both functional and non-functional requirements.
- To Understand design principles, modularity, and patterns, and apply function-oriented and object-oriented design methodologies.
- To Understand core testing concepts and techniques, and explore software reliability and reengineering processes.

Course Content	**TEACHING HOURS**
UNIT 1: Fundamentals of Software Engineering	**15 Hrs**
Concept and Nature of Software: Concept and Nature of Software, Software Crisis, Software Engineering – Concept, Goals and Challenges, Software Engineering Approach. Software Development Process, Process Models - Waterfall Model, Evolutionary and Throwaway Prototyping Model, Incremental and Iterative Models, Spiral Model, Agile Process Model, Component based and Aspect Oriented development Software Process and Project Measurement: Measures, Metrics and Indicators, Size - Oriented Metrics vs. Function - Oriented Metrics, Capability Maturity Model Integration (CMMI). COCOMO Model.	
UNIT 2: Requirements Engineering	**15 Hrs**
Introduction to Requirements Engineering - Why, What and Where. Requirements Types: functional and nonfunctional requirements. Requirement Engineering Framework. Requirement Elicitation Process and Techniques. Requirement Analysis and Modelling, Requirements prioritization, verification, and validation.	
UNIT 3: Design Engineering	**15 Hrs**
Basics of Design Engineering - Abstraction, Architecture, Patterns, Separation of concerns, Modularity, Functional Independence, refinement, Refactoring. Function oriented design, Design principles, Coupling and Cohesion, Design Notations & Specifications, Structured Design Methodology. Object-Oriented Design - Design Concepts, Design Methodology, Object-oriented analysis and design modeling using Unified Modeling Language (UML), Dynamic & Functional Modeling, Design Verification.	

UNIT 4: Software Testing and Reliability	**15 Hrs**
Software Testing – Concepts, Terminology, Testing & Debugging, Adequacy Criteria, Static vs. Dynamic Testing, Black Box vs. White Box Testing. Structural testing and its techniques. Functional Testing and its techniques, Mutation testing, Random Testing. Non-Functional Testing like Reliability, Usability, Performance and Security Testing. Introduction to Software Reliability: Basic Concepts, Correctness Vs Reliability, Software Reliability metrics, Operational Profile, Reliability Estimation and Predication, Reliability and Testing. Concept of Software reengineering, reverse engineering and change management.	

Textbooks

1. Shari Lawrence Pfleeger and Joanne M. Atlee - "Software Engineering: Theory and Practice," 4th Edition, Pearson, 2010.

Reference Books

1. Ian Sommerville - "Software Engineering," 10th Edition, Pearson, 2015.
2. Pankaj Jalote - "An Integrated Approach to Software Engineering," 3rd Edition, Narosa Publishing House, 2005.
3. Hans Van Vliet - "Software Engineering: Principles and Practice," 4th Edition, Wiley, 2016.
4. James F. Peters - "Software Engineering: An Engineering Approach," 1st Edition, Wiley & Sons, 2000.
5. Roger Pressman - "Software Engineering: A Practitioner's Approach," 8th Edition, McGraw-Hill Publications, 2014.

COURSE LEARNING OUTCOMES (CLO):

CLO1: Understand software engineering concepts, process models, and measurement techniques, and apply these to estimate and plan software development projects.
CLO2: Identify and analyze functional and non-functional requirements using requirement engineering processes and frameworks.
CLO3: Apply principles and practices of software design including structured and object-oriented approaches using UML and software design methodologies.
CLO4: Analyze and apply software testing techniques and reliability metrics for validating and verifying software quality and performance.

CLO-PLO Matrix for the Course

Unit-Wise CLOs	PLOs										
	1	2	3	4	5	6	7	8	9	10	Average (CLO)
MMCADSE125.1	3	3	3	2	2	1	2	2	2	3	**2.3**
MMCADSE125.2	3	3	3	2	3	1	2	2	2	3	**2.4**
MMCADSE125.3	3	3	3	2	3	1	2	3	2	3	**2.5**
MMCADSE125.4	3	3	3	2	3	1	2	2	2	3	**2.4**
Average (PLO)	**3.0**	**3.0**	**3.0**	**2.0**	**2.75**	**1.0**	**2.0**	**2.25**	**2.0**	**3.0**	**2.4**

To be effective from year-2025

DCE-II

To be effective from year-2025

COURSE TITLE: Advanced Database Systems

Course Code: MMCADDS125							**Examination Scheme**	**T**	**P**
Total number of Lecture Hours: 60							**External**	**72**	**-**
							Internal	**28**	**-**
Lecture (L):	**4**	**Practical (P):**	**-**	**Tutorial (T):**	**-**	**Total Credits**			**4**

Course Objectives

- To understand the foundational concepts of database systems, including data models, architecture, and ER modeling, in order to distinguish between data, information, and knowledge.
- To explore the relational data model and relational algebra operations, and to apply normalization techniques for designing efficient and consistent database schemas.
- To gain a conceptual understanding of distributed, parallel, and object-based database systems, their architectures, key features, and differences from traditional database models.
- To examine the principles of transaction management and recovery, including concurrency control techniques, serializability, and recovery methods to ensure database integrity and reliability.

Course Content	**TEACHING HOURS**
UNIT I: Introduction to Database Systems	**15 Hrs**
Introduction to Data, Information and Knowledge. Database basics – Need and evolution, Database and DBMS. Characteristics of Database Approach, Advantages and disadvantages of DBMS Approach. Database System Concepts and Architecture – Data Models, Schemas, and Instances, Database Models and Comparison, Three Schema Architecture and Data Independence. Database Languages and Interfaces. DBMS architectures. DBMS Classification. Data Modeling: Overview of Data Modeling, Entity-Relationship (ER) Modeling.	
UNIT II: Relational Data Model and Database Design	**15 Hrs**
Relational Data Model – Basic Concepts and Characteristics, Model Notation, Model Constraints and Database Schemas, Constraint Violations. Relational Algebra – basic concepts, Unary Relational Operations, Algebra Operations from Set Theory, Binary Operations, Additional Relational Operations. Criterion for Good Database Design. Database Design through Functional Dependencies & Normalization: Functional Dependencies, Lossless Join, Normal Forms: 1NF, 2NF, 3NF, BCNF.	
UNIT III: Overview of Distributed, Parallel, and Object-Based Databases	**15 Hrs**
Distributed Databases – Basic Concepts, Characteristics, and Design Issues, Data Fragmentation and Replication , Distributed Transparency: Location, Replication, and Fragmentation Transparency. **Parallel Databases** – Architecture Types (Shared Memory, Shared Disk, Shared Nothing), Concepts of Parallel Query Processing and Optimization. **Object-Based Databases** – Motivation, Features, and Architecture, Concepts of Object-Oriented Data Models and Object-Relational Databases, Comparison with Traditional Relational Databases	

UNIT IV: Transaction Management and Recovery	**15 Hrs**
Concepts of Transactions – ACID Properties, Transaction States, Issues in Concurrency – Lost Updates, Dirty Reads (Conceptual Examples), Serializability – Conflict and View Serializability (Overview Only), Overview of Concurrency Control Techniques – Two-Phase Locking, Timestamp Ordering, Database Recovery – Causes of Failure, Rollback Mechanisms, Recovery Approaches – Deferred and Immediate Update, Shadow Paging, Introduction to Logging and Checkpointing Techniques	

Textbooks:

1. Advanced Database Systems by Nabil R. Adam and Bharat K . Bhargava, ISBN 3 54057507-3 Springer-Verlag Berlin Heidelberg New York

Reference Books:

1. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 7thEdition, Pearson Education, 2017
2. Advanced Database Systems by Dr.John Kandiri
3. Abraham Silberschatz, Henry F. Korth, S.Sudarshan, "Database System Concepts", 6th Edition, 2014

COURSE LEARNING OUTCOMES (CLO):

CLO1: Explain database architecture, data models, and the advantages of the DBMS approach in organizing and managing structured data.

CLO2: Apply relational algebra operations and normalization rules to evaluate, optimize, and design database schemas that preserve integrity and consistency.

CLO3: Differentiate between centralized, distributed, parallel, and object-based databases, and describe their architectures, transparencies, and processing capabilities.

CLO4: Interpret and evaluate transaction management strategies and recovery techniques, ensuring database consistency and reliability in multi-user environments.

CLO-PLO Matrix for the Course

	PLO										
Unit-Wise CLOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Avg (CLO)**
MMCADDS125.1	3	2	2	1	2	1	1	2	2	2	**1.8**
MMCADDS125.2	3	2	2	1	3	1	1	1	2	2	**1.8**
MMCADDS125.3	3	3	2	1	3	1	1	2	1	3	**2.0**
MMCADDS125.4	3	3	3	1	3	2	1	2	1	3	**2.2**
Avg (PLO)	**3.0**	**2.5**	**2.3**	**1.0**	**2.8**	**1.3**	**1.0**	**1.8**	**1.5**	**2.5**	**1.97**

COURSE TITLE: Artificial Intelligence								
Course Code: MMCADAI125						**Examination Scheme**	**T**	**P**
Total number of Lecture Hours: 60						**External**	**72**	**-**
						Internal	**28**	**-**
Lecture(L):	**4**	**Practical (P):**	**-**	**Tutorial(T):**		**Total Credits**	**4**	

Course Objectives:

- To introduce the foundational concepts, historical evolution, knowledge representation methods, and intelligent agent models in Artificial Intelligence.
- To develop an understanding of fuzzy logic principles, inference techniques, and their application in handling imprecise or uncertain data.
- To equip students with various search and optimization techniques used in AI for problem-solving and decision-making.
- To explore inductive learning methods, uncertainty handling, and the fundamentals of artificial neural networks for pattern recognition and classification tasks.

Course Content	**TEACHING HOURS**
UNIT I:	**-15 Hrs**
Definition, history and applications AI. Representation of knowledge using logic-based approaches: propositional and first-order logic. Expert systems. Reasoning methods: Forward and backward chaining. Intelligent agents: types, structure, and environment interaction.	
UNIT II:	**- 15 Hrs**
Introduction to fuzzy logic and fuzzification. Linguistic terms, fuzzy sets, hedges, and operations. Fuzzy reasoning: Max-Min and Max-Product inferencing, multiple premise rules, Mamdani inference, aggregation, and defuzzification. Applications of fuzzy logic.	
UNIT III:	**-15 Hrs**
Search Algorithms – Uninformed search strategies, Informed search strategies, Hill Climbing, Constraint satisfaction problems, Optimization techniques: Genetic algorithms, Simulated annealing, Ant colony optimization, Swarm Particle optimization.	
UNIT IV:	**-15Hrs**
Inductive learning: categories, Rule extraction. Handling uncertainty in AI. Artificial Neural Networks (ANN): Basics of neural networks, architecture of perceptron and multilayer networks.	

Textbooks

1. *"***Artificial Intelligence: A Guide to Intelligent Systems"** by **Michael Negnevitsky, Latest Edition, 2020.**
2. *"Artificial Intelligence: A Modern Approach"* by Stuart Russell and Peter Norvig, **4th Edition,** 2020.
3. "Artificial Intelligence: A Guide for Thinking Humans" by Melanie Mitchell, **Latest Edition,** 2019

Reference Books

1. *"***Artificial Intelligence"** by **Elaine Rich, Kevin Knight, and Shivashankar B. Nair, 4th Edition, 2021.**
2. "Artificial Intelligence: Foundations of Computational Agents" by Michael Wooldridge, Ist Edition, 2021.
3. **"Nature-Inspired Optimization Algorithms"** by **Saeid Aziznejad, Gholamreza Z. Naderpour, and**

To be effective from year-2025

Mohammad A. H. Sadeghi, Ist Edition, 2019.

COURSE LEARNING OUTCOMES(CO):

CLO1: Identify and discuss various applications of AI across different domains and their impacts.
CLO2: Students will be able to explain the concepts of fuzzy logic including fuzzification and defuzzification.
CLO3: Implement and evaluate informed and uninformed search algorithms to solve problem-solving tasks.
CLO4: Students will be able to explain the principles of inductive learning and distinguish between different categories of inductive learning algorithms.

CLO-PLO Matrix for the Course

	PLO										
Unit-Wise CLOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Avg (CLO)**
MMCADAI125.1	3	1	2	2	2	1	1	2	1	2	**1.70**
MMCADAI125.2	3	2	2	1	2	1	1	2	1	2	**1.70**
MMCADAI125.3	3	2	2	1	2	1	1	1	1	2	**1.60**
MMCADAI125.4	2	3	3	1	3	1	2	3	2	3	**2.30**
Avg (PLO)	**2.75**	**2.00**	**2.25**	**1.25**	**2.25**	**1.00**	**1.25**	**2.00**	**1.25**	**2.25**	**1.82**

To be effective from year-2025

COURSE TITLE: Block Chain Technologies									
Course Code: MMCADBC125							**Examination Scheme**	**T**	**P**
Total number of Lecture Hours: 60							**External**	**72**	-
							Internal	**28**	-
Lecture (L):	**4**	**Practical (P):**	-	**Tutorial (T):**	-		**Total Credits**	**4**	

Course Objectives:

- To explain the foundational concepts of blockchain technology, including its structure, cryptographic principles, consensus mechanisms, and major blockchain platforms.
- To Analyze blockchain network components, security mechanisms, and real-world use cases, with a focus on the architecture and functioning of cryptocurrencies like Bitcoin and Ethereum.
- To Develop and deploy smart contracts and decentralized applications (DApps) using appropriate blockchain development tools, languages, and environments.
- To Evaluate emerging trends and interdisciplinary applications of blockchain in areas such as IoT, AI, big data, and quantum computing, and assess the potential of future technologies like Web 3.0 and decentralized identity..

Course Content	**TEACHING HOURS**
Unit 1: Introduction to Blockchain Technology	**15 Hrs**
Introduction to Blockchain - Definition, History, and Evolution. Basic Concepts - Distributed Ledger Technology (DLT), Cryptography, and Consensus Mechanisms. Types of Blockchains - Public, Private, Consortium, and Hybrid Block chains. Blockchain Structure - Blocks, Chains, Nodes, and Transactions. Cryptographic Foundations - Hash Functions, Digital Signatures, Public and Private Keys. Consensus Algorithms - Proof of Work (PoW), Proof of Stake (PoS), Delegated PoS. Smart Contracts - Definition, Creation, Execution, and Security Issues. Overview of Major Blockchain Platforms - Bitcoin, Ethereum, Hyperledger.	
Unit 2: Blockchain and Cryptocurrencies	**15 Hrs**
Blockchain Networks - Nodes, Peer-to-Peer Networks, and Distributed Consensus. Security in Blockchain - Threats, Attacks, and Countermeasures. Blockchain Use Cases - Financial Services, Supply Chain, Healthcare. Introduction to Cryptocurrencies - Bitcoin and Altcoins. Bitcoin Architecture - Blockchain, Mining, Wallets, and Transactions. Ethereum and Smart Contracts - Solidity, DApps, and Gas. Cryptocurrency Wallets - Types, Security, and Key Management.	
Unit 3: Blockchain Development and Implementation	**15 Hrs**
Introduction to Blockchain Development - Tools, Platforms, and IDEs. Blockchain Development Languages - Solidity, Vyper, Go, and JavaScript. Building Smart Contracts - Basics, Writing, and Deploying.	

To be effective from year-2025

Developing DApps - Frontend, Backend, and Smart Contract Integration. Ethereum Development Environment - Truffle, Ganache, Remix. Hyperledger Fabric - Architecture, Components, and Development. Testing Blockchain Applications - Unit Tests, Integration Tests .	
Unit 4: Advanced Topics and Future Directions in Blockchain	**15 Hrs**
Blockchain in IoT - Use Cases, Challenges, and Solutions. Blockchain and Big Data - Integration, Analytics, and Use Cases. Blockchain in AI - Synergies, Applications, and Challenges. Blockchain and Cloud Computing - Decentralized Cloud Solutions. Green and Sustainable Blockchain Technologies. Quantum Computing and its Impact on Blockchain . Future Directions - Web 3.0, Decentralized Identity, and Tokenization of Assets.	

Textbooks:

1. "Blockchain Technology: Concepts and Applications" by Kumar Saurabh and Ashutosh Saxena, McGraw-Hill Education (2020).
2. "Cryptocurrency and Blockchain Technology" by Shaik Nasrullah and M. Balamurugan, Pearson (2021).
3. "Blockchain and Cryptocurrency" by B. B. Gupta and Hemraj Saini, PHI Learning (2020).

Reference Books:

1. "Cryptography and Blockchain Technology" by Atul Kahate, McGraw-Hill Education (2018).
2. "Blockchain: Principles and Applications" by Umesh Kumar Singh and Kavita Rani, Pearson (2020).
3. "Blockchain Technology and Applications" by M. S. Kiruthika and B. Prabu, PHI Learning (2021).

COURSE LEARNING OUTCOMES (CLO):

CLO1: Understand the fundamental concepts, cryptographic principles, types, and architectures of blockchain systems and analyze major blockchain platforms.
CLO2: Evaluate the structure, operations, and applications of cryptocurrencies and identify the security implications in blockchain networks.
CLO3: Apply blockchain programming languages and development frameworks to build and deploy smart contracts and decentralized applications (DApps).
CLO4: Apply blockchain programming languages and development frameworks to build and deploy smart contracts and decentralized applications (DApps).

LEVEL OF CLO-PLO MAPPING TABLE

	PLOs										
Unit wise CLOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Average (CLO)**
MMCADBC125.1	3	2	2	2	3	2	2	2	2	3	**2.3**
MMCADBC125.2	3	3	2	2	3	2	2	2	3	3	**2.5**
MMCADBC125.3	3	3	3	2	3	1	2	3	2	3	**2.5**
MMCADBC125.4	3	2	2	2	3	2	3	3	3	2	**2.5**

To be effective from year-2025

Average (PLO)	**3.0**	**2.5**	**2.25**	**2.0**	**3.0**	**1.75**	**2.25**	**2.5**	**2.5**	**2.75**	**2.45**