SDGI GLOBAL UNIVERSITY

PROGRAMMESTRUCTURE

SCHOOLOFCOMPUTER APPLICATIONS

Scheme & Syllabus

of

MasterofComputerApplications(MCA) 2

Years Full time program

Specialization-ArtificialIntelligenceandDataScience

Academic Program

W.E.F 2024

SDGI Global University, Ghaziabad(U P)

VisionoftheUniversity

To be recognized as an Institution of excellence, facilitating learning, fostering creativity, knowledge creation, innovations, consultancy and leadership in multiple areas to build a conscious community that will positively impact living beings for a sustainable future.

MissionoftheUniversity

1. To Create conducive environment for an interactive and application oriented experiential learning making the Institute a preferred destination for work and study.
2. To Foster creativity, research and innovation orientation in students and faculty in basic and applied areas in all of its disciplines, provide cost effective solutions and nurture entrepreneurial capabilities to accelerate growth.
3. To act as a catalyst in social change by developing academic, social, political, technological, scientific, industrial and business leadership in the spirit "Think Globally and Act Locally"; by providing ample opportunities to develop team spirit, sportsmanship and love for culture and national heritage.

Core Values

1. **Integrity**
2. **Honesty**
3. **Transparency**
4. **Empathy**

School of Computer Applications

Vision of School

To be a premier institution in computing sciences, recognized for pioneering research, transformative education, and impactful contributions to society, shaping the future of technology and driving positive.

Mission of School

To empower students with cutting-edge knowledge and skills in computing sciences, foster a culture of innovation, and prepare them to address the challenges of a rapidly evolving digital world through rigorous academics, experiential learning, and interdisciplinary collaboration.

CoreValues

1. Excellence
2. Innovation
3. Sustainability
4. Global Perspective

Program Educational Objectives (PEOs)

1. Technical Excellence: Prepare graduates with a strong foundation in computing principles, advanced technical skills, and the ability to apply these skills to solve complex problems in various domains of computing.

2. Professional Development: Equip graduates with the necessary skills to succeed in professional careers, including software development, systems analysis, and IT management, while fostering lifelong learning and adaptability to emerging technologies**.**

3. Research and Innovation: Encourage graduates to engage in research and innovation, contributing to the advancement of technology and knowledge in the field of computer science.

4. Ethical and Social Responsibility: Instill a sense of professional ethics, responsibility, and awareness of the social and environmental impact of technology, ensuring that graduates make informed and ethical decisions in their careers**.**

5. Communication and Teamwork: Develop graduates' abilities to communicate effectively and work collaboratively in multidisciplinary teams, enhancing their effectiveness in professional and academic environments.

Program Outcomes (POs)

1. Knowledge and Understanding: Demonstrate a comprehensive understanding of computing fundamentals, including programming, software engineering, database management, and network systems.

2. Problem-Solving Skills: Apply analytical and problem-solving techniques to design, develop, and implement solutions for complex computing problems.

3. Research and Development: Engage in research and development activities, including the ability to conduct experiments, analyze data, and contribute to technological advancements.

4. Professional Ethics and Responsibility: Adhere to professional ethics and legal standards in the practice of computing, with an understanding of the broader impact of technology on society**.**

5. Communication and Teamwork: Exhibit effective communication skills and the ability to collaborate with peers, stakeholders, and clients in professional and academic settings.

Program Specific Outcomes (PSOs)

1. Advanced Computing Techniques: Demonstrate proficiency in advanced computing techniques, including the development and implementation of complex software systems, data analytics, and machine learning.

2. Software Development Lifecycle: Apply knowledge of the software development lifecycle to manage projects, from requirements gathering and design to implementation, testing, and maintenance.

3. Emerging Technologies: Stay updated with emerging technologies and industry trends, including cloud computing, cybersecurity, and artificial intelligence, and apply this knowledge to solve real-world problems.

4. Research and Innovation Skills: Develop and apply research skills to contribute to innovative solutions in computing, including the ability to design and conduct experiments, analyze results, and publish findings.

5. Professional and Ethical Practice: Exhibit professional conduct and ethical practices in all aspects of computing, including adherence to industry standards, legal regulations, and societal responsibilities.

These PEOs, POs, and PSOs align with UGC norms by ensuring that the MCA program provides a well-rounded education that prepares students for various aspects of their professional and academic careers.

Multiple Exit and Multiple Entry Rule:

According to the UGC (University Grants Commission) guidelines for a PG program with multiple entry and exit points, the credit system typically includes:

1. Total Credits for a PG Program: Generally ranges from 60 to 120 creditsRegards for a full Master's program, depending on the duration (usually 2 years).

2. Credits for Different Entry/Exit Points.

Exit after 1 Year (or 30-40 credits): Often allows students to earn a Postgraduate Diploma or a similar qualification.

Exit after 2 Years (or 60-80 credits): Typically results in the completion of the full Master's degree.

Complete the Program (or 90-120 credits): This might be for programs that have extended components or specializations.

These credits are structured to ensure that students can enter, exit, or re-enter the program at different stages while still gaining recognized qualifications. The exact credit requirements can vary based on the specific UGC guidelines adopted by the institution.

TotalCredits forthe Batch 2024= 81Credits

SEMESTERWISECREDITS				CREDITS
I	II	III	IV	Total
19	21	23	18	81

CC - Core Course
OE - Open Elective/ Multidisciplinary
AE - Ability Enhancement Compulsory Course
VAC - Value Added Course
SE - Skill Enhancement
DSC-Discipline specific course

AI-DSSpecializationsSubjects(MCA)							
Semester	Code	CourseName	L (Hr.)	T (Hr.)	P (Hr.)	Credits	Type
I	M020224102	Introduction to Artificial Intelligence Machine LearningandDataScience	4	0	2	5	DSC (Minor)
II	M020224205	Python for Artificial IntelligenceandData Science	4	0	2	5	DSC (Minor)
II	M020224204	AdvanceDataScience	4	0	2	5	DSC (Minor)
III	M020224303	Advance Linear Algebra ProbabilityandStatistics for Machine Learning	4	0	0	4	DSC (Minor)
III	M020224302	Deep Learning	4	0	2	5	DSC Minor
Total			20	0	8	24	

Bridge Course(1st Semester)					
I	MSGUBR2401	Fundamental of Information Technology	30 Hrs		Bridge
I	MSGUBR2402	Programming in C	30Hrs		Bridge

SDGI GLOBAL UNIVERSITY, GHAZIABAD (SGU)

Semester–Wise Teaching Scheme PG Program

Programme: MCA with AI - ML

Semester – I

Batch – 2024 – 26

Academic year: 2024 – 25

S. No	Status	Paper Code	Subjects	Study Scheme Lec / Week			Hours	Credits	CIE	ESE	Total	Pass Marks
				L	T	P						
1	CC1	**M020224101**	CompetitiveProgramming (Using C++)	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
3	CC2	**M020224102**	AdvanceData Structures and Algorithms	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
4	CC3	**M020224103**	AdvanceOperatingSystem	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
5	CC Lab	**M020224151**	CompetitiveProgramming (Using C++) Lab	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
6	CC Lab	**M020224152**	AdvanceDataStructures and Algorithm Lab	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
7	**CC Lab**	**M020224153**	AdvanceOperatingSystem Lab	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
7	**DSC-1**	**M020224104**	**IntroductiontoArtificial Intelligence Machine Learning and Data Science (Minor)**	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
Total				**16**	**0**	**6**	**22**	**19**	**380**	**320**	**700**	**280**

CC - Core Course
OE - Open Elective/ Multidisciplinary
AE - Ability Enhancement Compulsory Course
VAC - Value Added Course
SE - Skill Enhancement
DSC-Discipline specific course

SDGI GLOBAL UNIVERSITY, GHAZIABAD (SGU)

Semester–Wise Teaching Scheme PG Program

Programme: MCA with AI - ML

Semester – II

Batch – 2024 – 26

Academic year: 2024 – 25

S. No	Status	Paper Code	Subjects	Study Scheme Lec / Week			Hours	Credits	CIE	ESE	Total	Pass Marks
				L	T	P						
1	**CC-4**	**M020224201**	Advance Database ManagementSystem	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
2	CC-5	**M020224202**	AdvanceJava	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
3	CC Lab	**M020224251**	Advance Database Management Systems lab	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
4	CC Lab	**M020224252**	AdvanceJavaLab	**0**	**0**	**4**	**4**	**2**	**60**	**40**	**100**	**40**
5	**DSC-2**	**M020224203**	**Artificial Intelligence Search Method for Problem Solving (Minor)**	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
6	DSC-3	**M020224204**	**Data Analytics with Python (Minor)**	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
7	DSC Lab	**M020224253**	**Artificial Intelligence Search Method for Problem Solving Lab (Minor)**	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
8	**DSC Lab**	**M020224254**	**Data Analytics with Python Lab (Minor)**	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
				16	**0**	**10**	**26**	**21**	**440**	**360**	**800**	**320**

CC - Core Course
OE - Open Elective/ Multidisciplinary
AE - Ability Enhancement Compulsory Course
VAC - Value Added Course
SE - Skill Enhancement
DSC-Discipline specific course

SDGI GLOBAL UNIVERSITY, GHAZIABAD (SGU)

Semester–Wise Teaching Scheme PG Program

Programme: MCA with AI - ML

Semester – III

Batch – 2024 – 26

Academic year: 2025 – 26

S. No	Status	Paper Code	Subjects	Study Scheme Lec / Week			Hours	Credits	CIE	ESE	Total	Pass Marks
				L	T	P						
1	CC-6	**M020224301**	WebTechnology	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
2	CC Lab	**M020224351**	WebTechnologyLab	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
3	CC-13	**M020224352**	Project	**0**	**0**	**8**	**8**	**4**	**60**	**40**	**100**	**40**
4	DSC-4	**M020224303**	**AdvanceLinearAlgebra Probability and Statistics for Machine Learning**	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
5	**DSC-5**	**M020224304**	**Deep Learning**	**4**	**0**	**0**	**4**	**4**	**50**	**50**	**100**	**40**
6	DSC Lab	**M020224354**	**DeepLearning Lab**	**0**	**0**	**2**	**2**	**1**	**60**	**40**	**100**	**40**
7	AEC-1	**BSGUAE2405**	Research Writing Skill	**2**	**0**	**0**	**2**	**2**	**50**	**50**	**100**	**40**
8	**VAC-1**	**BSGUVA2402**	**Human Values and Professional Ethics**	**3**	**0**	**0**	**3**	**3**	**50**	**50**	**100**	**40**
Total				**17**	**0**	**12**	**29**	**23**	**430**	**370**	**800**	**320**

CC - Core Course
OE - Open Elective/ Multidisciplinary
AE - Ability Enhancement Compulsory Course
VAC - Value Added Course
SE - Skill Enhancement
DSC-Discipline specific course

CompetitiveProgramming (Using C++)

Program:MCA-I **L T P C**

Course Code:M020224101 **4 0 0 4**

Course Objective:
Introduces Object Oriented Programming concepts using the C++ language.
Introduces the principles of data abstraction, inheritance and polymorphism
Introduces the principles of virtual functions and polymorphism
Introduces handling formatted I/O and unformatted I/O
Introduces exception handling

CO	Course Outcome	Bloom's Level
CO1	Introduces Object Oriented Programming concepts using the C++ language.	**K1**
CO2	Understanding the principles of data abstraction, inheritance, and polymorphism.	**K2**
CO3	Apply the principles of virtual functions and polymorphism.	**K3**
CO4	Understanding the concept of Memory management.	**K4**
CO5	Understand the Exception Handling	**K2**

Detailed Syllabus

Unit I

Object-Oriented Thinking: Different paradigms for problem solving, need for OOP paradigm, differences between OOP and Procedure oriented programming, Overview of OOP conceptsAbstraction, Encapsulation, Inheritance and Polymorphism. C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References. Flow control statement- if, switch, while, for, do, break, continue, goto statements. Functions - Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions. Dynamic memory allocation and de-allocation operators-new and delete, Preprocessor directives.

Unit II

C++ Classes and Data Abstraction: Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

Unit III

Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class. Virtual Functions and Polymorphism: Static and Dynamic binding, virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.

Unit IV

C++ I/O: I/O using C functions, Stream classes hierarchy, Stream I/O, File streams and String streams, Overloading operators, Error handling during file operations, Formatted I/O.

Unit V

Exception Handling: Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions.

Text Books:

1. Object Oriented Programming with C++ by E. Balagurusamy, McGraw-Hill Education (India)
2. Turbo C++ by Robert Lafore, Pearson Education

Reference Books:

1. Big C++ - Wiley India
2. C++: The Complete Reference- Schildt, McGraw-Hill Education (India)

Advance Data Structures and Algorithms

Program:MCA-I **L T P C**

Course Code:M020224102 **4 0 0 4**

Course Objective:
Analyze time and space complexities for optimal problem-solving.
Implement and compare AVL, B-trees, and red-black trees.
Apply heaps and binomial queues for optimization tasks.
Design efficient graph algorithms for shortest paths and sorting.
Explore disjoint sets and string matching for connectivity and pattern recognition.

CO	Course Outcome	Bloom's Level
CO1	**Understand the basic principles and operations of data structures.**	**K2**
CO2	**Apply Hashing, Disjoint sets and String Matching techniques for solving problems effectively.**	**K3**
CO3	**Apply the concepts of advanced Trees and Graphs for solving problems effectively.**	**K3**
CO4	**Analyze the given scenario and choose appropriate Data Structure for solving problems.**	**K4**
CO5		

Detailed Syllabus

Unit-1

Hashing–
GeneralIdea,HashFunction,SeparateChaining,HashTableswithoutlinkedlists:LinearProbing,QuadraticProbing,DoubleHashing,Rehashing, Hash Tables in the Standard Library, Universal Hashing, ExtendibleHashing.

Unit-2

PriorityQueues(Heaps)–Model,Simpleimplementations,BinaryHeap:Structure Property, Heap Order Property, Basic Heap Operations: insert, delete,Percolatedown,Other Heap Operations.

BinomialQueues:BinomialQueueStructure,BinomialQueueOperations,ImplementationofBinomialQueue, PriorityQueuesin theStandard Library.

Unit-3

Trees–AVL:SingleRotation,DoubleRotation,B-Trees. Multi-waySearchTrees–2-3Trees:SearchingforanElementina2-3Tree,InsertingaNewElementina 2-3 Tree,Deletingan Element froma 2-3 Tree. Red-BlackTrees–Propertiesofred-blacktrees,Rotations,Insertion,Deletion.

Unit-4

Graphs Algorithms – Elementary Graph Algorithms: Topological sort, SingleSource Shortest Path Algorithms: Dijkstra's, Bellman-Ford, All-Pairs ShortestPaths:Floyd-Warshall's Algorithm.

Unit-5

DisjointSets–Equivalencerelation,BasicDataStructure,SimpleUnionandFindalgorithms, Smart Unionand Path compression algorithm. StringMatching–Thenaivestring-matchingalgorithm,TheRabin-Karpalgorithm,TheKnuth-Morris-Pratt algorithm.

Test Book: 1.DataStructuresandAlgorithmAnalysisinC++, MarkAllenWeiss,4th Edition,2014,Pearson.IntroductiontoAlgorithms,ThomasHCormen,CharlesE.Leiserson,RonaldL.Rivest,CliffordStein,3rdEdition, 2009,TheMIT Press.

Refrence Book: FundamentalsofComputerAlgorithms,EllisHorowitz,SatrajSahaniandRajasekharam,2ndEdition,

2009, UniversityPress Pvt.Ltd.

Advance operating System

Program:MCA-I **L T P C**

Course Code:M020224103 **4 0 0 4**

Course Objective:
Understand advanced concepts of distributed systems and architectures.
Analyze and apply process synchronization and deadlock handling techniques.
Explore distributed file systems and memory management in distributed environments.
Learn fault tolerance, security, and recovery mechanisms in distributed systems.
Gain expertise in cloud computing, virtualization, and the design of scalable systems.

CO	Course Outcome	Bloom's Level
CO1	**Understand the architecture and synchronization mechanisms of distributed systems.**	**K2**
CO2	**Analyze and handle deadlock conditions in advanced operating systems.**	**K4**
CO3	**Manage memory and file systems in a distributed environment.**	**K3**
CO4	**Implement fault tolerance, security, and recovery protocols in operating systems.**	**K3**
CO5	**Explore cloud computing technologies and virtualization techniques.**	**K2**

Detailed Syllabus

Unit-1

Distributed Systems and ArchitecturesOverview of Distributed Systems: Definition, goals,and types of distributed systems.Architectures: Client-server, peer-to-peer, and multi-tier architecture.Communication: Remote Procedure Calls (RPC), Remote Method Invocation (RMI), message-oriented communication.Synchronization in Distributed Systems: Clock synchronization, Lamport's logical clock, vector clocks.Case Studies: Distributed file systems, examples from Hadoop or Google File System.

Unit 2:

Process Synchronization and DeadlocksAdvanced Process Synchronization: Mutual exclusion, synchronization primitives (semaphores, monitors), and classical synchronization problems.Deadlock Detection, Avoidance, and Prevention: Concepts of deadlock, conditions, deadlock handling techniques.Resource Allocation Graphs: Detection algorithms and Banker's algorithm.Real-Time Operating Systems (RTOS): Scheduling algorithms, real-time task synchronization.

Unit 3:

Distributed File Systems and Memory ManagementDistributed File Systems (DFS): Design and implementation, consistency and replication, fault tolerance, NFS and HDFS.Memory Management in Distributed Systems: Distributed shared memory, consistency models, pagereplacement algorithms.Advanced Paging Techniques: Demand paging, pre-paging, page replacement policies.
Virtual Memory Systems: Segmentation and paging, translation lookaside buffer (TLB).

Unit 4:

Fault Tolerance, Security, and Recovery in Distributed SystemsFault Tolerance Mechanisms: Redundancy, replication, failover techniques.Consensus Algorithms: Paxos, Raft.Security in Distributed Systems: Authentication, encryption, secure communication protocols, Kerberos.Recovery and Checkpointing: Failure recovery mechanisms, checkpointing algorithms, rollback recovery.

Unit 5:

Cloud and VirtualizationCloud Computing Concepts: Virtualization, cloud architecture, cloud services (IaaS, PaaS, SaaS).Virtual Machines and Hypervisors: Types of virtualization (full, para, hardware-assisted), hypervisor types (Type-1, Type-2).Containerization: Docker, Kubernetes, microservicesarchitecture.Resource Management in Cloud Systems: Scheduling, load balancing, dynamic resource allocation.Case Studies: VMware, Amazon Web Services (AWS), Google Cloud Platform (GCP).

Text book:**1.**Distributed Systems: Principles and Paradigms by Andrew S. Tanenbaum and Maarten

Van Steen 2.Modern Operating Systems by Andrew S. Tanenbaum 3.Distributed Operating **Systems** by MukeshSinghal and Niranjan G. Shivaratri.
Reference Book:Real-Time Systems: Design Principles for Distributed Embedded Applications by Hermann Kopetz.

CompetitiveProgramming (Using C++) Lab

Program:MCA-I **L T P C**

Course Code:M020224151 **0 0 21**

Course Objective:
Students will demonstrate the ability to utilize variables and control flow statements, design functions with parameter passing techniques and default arguments, implement classes with constructors and destructors, develop class hierarchies using inheritance and polymorphism, and manage file handling along with exception management in C++.

CO	Course Outcome	Bloom's Level
CO1	Demonstrate the use of variables, arithmetic operations, and control flow statements inC++.	**K3**
CO2	Illustrate parameter passing techniques and default arguments in function design.	**K2**
CO3	Implement classes with constructors and destructors to model real-world entities.	**K3**
CO4	Create class hierarchies and use virtual functions for dynamic behavior..	**K2**
CO5	Utilize file handling and exception management for robust program development..	**K3**

List of experiments.

1. Write a simple C++ program that declares variables of different data types, performs arithmetic operations, and outputs the results.
2. Create a program that uses `if`, `switch`, `for`, `while`, and `do-while` statements to perform calculations based on user input.
3. Write a program with various functions, demonstrating different parameter passing techniques (by value, by reference) and the use of default arguments.
4. Define a class representing a simple bank account, including member functions for deposit, withdrawal, and checking the balance. Implement constructors and destructors.
5. Create a base class `Shape` and derived classes `Circle` and `Rectangle`. Implement area and perimeter calculations using overridden functions.
6. Use virtual functions to implement a system where derived classes (`Circle`, `Rectangle`) can override a base class function for calculating area.
7. Write a program that reads data from a file, processes it (e.g., calculates the average of numbers), and writes the result to another file. Include error handling for file operations.
8. Implement a program that performs division operations and handles potential division-by-zero exceptions. Demonstrate throwing and catching exceptions with proper error messages
9. Write a program that dynamically allocates memory for an array of integers, allows user input, and displays the array. Ensure proper memory deallocation using `delete`

Advance Data Structure and Algorithm lab

Program:MCA-I **L T P C**

Course Code:M020224152 **0 0 2 1**

Course Objective:
Implement AVL Trees and Red-Black Trees to understand their balance maintenance mechanisms.
Perform insert, delete, and search operations in both trees to evaluate their performance.
Compare and analyze the balance maintenance and performance of AVL and Red-Black Trees in various scenarios

CO	Course Outcome	Bloom's Level
CO1	Demonstrate the ability to implement and manipulate advanced balanced binary search trees, including AVL and Red-Black Trees.	**K3**
CO2	Evaluate and compare the efficiency of different tree balancing techniques in terms of operation time complexity.	**K5**
CO3	alyze the impact of tree balancing on performance for dynamic operations such as insertion, deletion, and search.	**K4**
CO4	Develop skills in designing algorithms that ensure balanced tree structures for optimal performance.	**K6**
CO5	Understand and explain the trade-offs between AVL Trees and Red-Black Trees in practical scenarios.	**K2**

List of experiments

1. Insert, delete, and search operations. Analyze the balance maintenance and performance of both trees.
2. Compute shortest paths in a graph with non-negative weights (Dijkstra) and with possible negative weights (Bellman-Ford). Compare their performance and correctness.
3. Create a graph and compute the MST using both algorithms. Compare their efficiency and output.
4. Handle collisions and measure performance metrics such as load factor and retrieval time.
5. Insert and search for strings. Implement additional functionalities like prefix matching and autocomplete.
6. Implement algorithms to construct suffix trees/arrays and use them for various string processing problems.
7. Perform range sum queries and range updates, demonstrating the efficiency gains from lazy propagation.
8. Insert, delete, and search operations. Compare the properties and performance of B-Trees vs. B+ Trees.
9. Choose one advanced algorithm, implement it, and solve a relevant problem. Analyze its efficiency and applications.
10. Solve problems and analyze the time and space complexity of the dynamic programming approaches.

Advance operating System lab

Program:MCA-I **L T P C**

Course Code:M020224153 **0 0 2 1**

Course Objective:
Configure a basic distributed system across multiple machines, ensuring network connectivity and communication between nodes.
Develop a simple client-server system where the client sends requests, and the server processes and responds using socket programming.
Create a simple peer-to-peer file sharing application where nodes communicate directly to exchange data without a centralized server.
Build a multi-tier web application with separate presentation, application, and data layers using web technologies.
Implement an RPC mechanism where a client can call functions on a server located on a different machine.

CO	Course Outcome	Bloom's Level
CO1	**Understand the architecture, goals, and types of distributed systems.**	**K2**
CO2	**Gain practical understanding of client-server communication.**	**K3**
CO3	**Understand the working of peer-to-peer architecture in distributed systems.**	**K2**
CO4	**Learn the structure and benefits of multi-tier architectures.**	**K1**
CO5	**Learn the mechanism for procedure calls in distributed environments.**	**K4**

List of experiments

1. Introduction to Distributed Systems: Setting Up a Basic Distributed System
2. Implementing a Client-Server Architecture using Sockets
3. Peer-to-Peer Architecture: File Sharing System
4. Multi-Tier Architecture: Web Application using Three Tiers (Presentation, Logic, Data)
5. Remote Procedure Calls (RPC) Implementation
6. Message-Oriented Communication: Implementing a Message Queue
7. Clock Synchronization in Distributed Systems: Lamport's Logical Clock
8. **Vector Clock Implementation for Event Causality**
9. 9.**Case Study: Hadoop Distributed File System (HDFS)**

IntroductiontoArtificial Intelligence Machine Learning and Data Science

Program:MCA-I **L T P C**

Course Code:M0202241044 0 04

Course Objective:
Understand the scope, branches, and history of AI.
Learn the fundamentals of machine learning and key algorithms
Gain practical skills in data preprocessing and advanced ML techniques
Explore data science concepts and tools for data analysis
Study advanced AI applications like NLP, big data, and current trends.

CO	Course Outcome	Bloom's Level
CO1	**Understand the history and societal impacts of AI..**	**K2**
CO2	**Apply and evaluate machine learning algorithms**	**K3**
CO3	**Use data preprocessing and optimize models effectively**	**K3**
CO4	**Analyze and visualize data using popular libraries**	**K4**
CO5	**Solve real-world problems using advanced ML and AI techniques**	**K6**

DetailedSyllabus:

Unit-1

IntroductiontoArtificialIntelligence

Definition and scope of Artificial Intelligence, Historical background and milestonesin AI development, Various branches of AI: symbolic AI, statistical AI, etc., ApplicationsofAIindifferentfieldslikehealthcare,finance,gaming,etc.,Ethical considerationsandsocietalimpactofAI

Unit-2

FundamentalsofMachineLearning

Introduction to Machine Learning and its importance, Types of Machine Learning: Supervised, Unsupervised, and Reinforcement Learning, Basic concepts: features, labels, training data, etc., Popular Machine Learning algorithms: Linear Regression, LogisticRegression,DecisionTrees,k-NearestNeighbors,etc.,EvaluationmetricsforMachineLearningmodels:accuracy,precision,recall,F1-score,etc.

Unit-3

MachineLearning Techniques

Data preprocessing techniques: handling missing data, feature scaling, feature encoding, etc., Model selection and hyperparameter tuning, Cross-validation techniques,Ensemblemethods:Bagging,Boosting,RandomForests,etc.,Introductiontodeeplearningandneuralnetworks

Unit-4

IntroductiontoDataScience

What is Data Science and why it is important?, Role of Data Scientist and skills required, Data acquisition: sources of data, data formats, data cleaning, etc., ExploratoryDataAnalysis(EDA):statisticalanalysis,datavisualizationtechniques,Introductiontolibraries/tools:NumPy,Pandas,Matplotlib,Seaborn,etc

Unit-5

AdvancedTopicsandApplications

Advanced Machine Learning techniques: Support Vector Machines (SVM), Neural Networks, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), etc., Natural Language Processing (NLP) and its applications, Introduction to BigDatatechnologies:Hadoop,Spark,etc.,Casestudiesandreal-worldapplications invariousdomains,FuturetrendsandcareerprospectsinAI,ML,andData Science

Text Books:

1. "Artificial Intelligence:AModernApproach"byStuartRussellandPeterNorvig
2. "IntroductiontoMachineLearning"byEthemAlpaydin

Advance Database ManagementSystem

Program:MCA-II **L T P C**

Course Code:M020224201 **4 0 0 4**

Course Objective:
Understand and apply complex Entity-Relationship (ER) diagrams and advanced ER modeling concepts.
Design and utilize object-oriented database models, including concepts like classes, objects, inheritance, and polymorphism.
Model and query time-varying data using temporal database models.
Implement advanced SQL features and procedural extensions (PL/SQL, T-SQL) to handle complex queries.
Use recursive queries to solve complex data retrieval challenges.

CO	Course Outcome	Bloom's Level
CO1	Create and interpret complex ER diagrams and advanced database models.	**K2**
CO2	Design and query object-oriented databases effectively using advanced object-oriented concepts.	**K2**
CO3	Model, query, and manage time-varying data in temporal databases.	**K3**
CO4	Write and optimize advanced SQL queries using procedural extensions and recursive techniques.	**K4**
CO5	Apply advanced querying techniques to solve complex data retrieval problems effectively	**K5**

Detailed Syllabus

Unit 1

Advanced Data Models and Query Languages: Entity-Relationship Model: Complex ER diagrams, advanced concepts in ER modeling, and extensions.**Object-Oriented Database Models**: Object-oriented concepts, classes, objects, inheritance, and polymorphism in databases.**Temporal Databases**: Time-varying data, temporal data models, and querying temporal data.**Query Languages**: Advanced SQL features, procedural extensions (PL/SQL, T-SQL), and querying with recursion.

Unit 2

Database Optimization and Performance Tuning: Query Optimization: Cost-based optimization, query execution plans, and optimization techniques.**Indexing**: Advanced indexing techniques (B-trees, B+-trees, bitmap indexes), and index optimization.**Database Design**: Normalization and denormalization, schema design, and performance considerations.**Caching and Buffer Management**: Strategies for caching, buffer management, and optimization of database performance.

Unit 3

Distributed Databases: Distributed Database Models: Architecture, data distribution, and replication strategies.**Concurrency Control**: Techniques for managing concurrent transactions, locking mechanisms, and conflict resolution.**Distributed Query Processing**: Techniques for query processing in distributed databases, and optimization strategies.**Distributed Transactions**: Two-phase commit protocol, transaction management, and consistency models.

Unit 4

NoSQL and NewSQL Databases: NoSQL Databases: Types (document, column-family, key-

value, graph), CAP theorem, and use cases.**NewSQL Databases**: Features of NewSQL systems, comparison with traditional SQL and NoSQL, and examples.**Data Modeling**: Designing schemas for NoSQL and NewSQL databases.**Querying and Indexing**: Techniques for querying and indexing in NoSQL and NewSQL databases.

Unit 5

Advanced Topics and Applications: Big Data Integration: Integration of databases with big data technologies (Hadoop, Spark).**Data Warehousing and OLAP**: Concepts of data warehousing, OLAP cubes, and data warehouse design.**Database Security**: Security models, access control, encryption, and auditing.**Blockchain and Databases**: Introduction to blockchain technology, its integration with databases, and practical applications.

Text Book:1.Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan

2.Fundamentals of Database Systems" by RamezElmasri and Shamkant B. Navathe.

Reference Book:1."Advanced Database Systems" by Carlo Zaniolo, Stefano Ceri, and Christos Faloutsos.

3. "Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke

AdvanceJava

Program:MCA-II **L T P C**

Course Code:M020224202 **4 00 4**

Course Objective:
To enhance knowledge of object-oriented programming using Java.
To introduce advanced topics like networking, multi-threading, and database connectivity
To develop web applications using servlets and JSP.
To provide a solid foundation in enterprise-level Java technologies

CO	Course Outcome	Bloom's Level
CO1	Develop applications using advanced object-oriented features of Java..	**K3**
CO2	Create robust applications using Java Networking and Multi-threading	**K2**
CO3	Implement database connectivity using JDBC	**K3**
CO4	Build dynamic web applications using Servlets and JSP	**K4**
CO5	Understand the basics of Java frameworks like Spring and Hibernate	**K5**

Detailed Syllabus

Unit 1:

Advanced Object-Oriented Concepts: Generics and Collections Framework, Exception Handling (Advanced), Java Input/Output (I/O) System, Java Annotations and Reflection,Java 8 Features: Lambdas and Streams API.

Unit 2:

Multithreading and Concurrency: Multithreading and Thread Lifecycle, Thread Synchronization and Communication, Concurrency API: Executor Framework, Callable, Future, Fork-Join Framework, Java Memory Model.

Unit 3:

Java Networking: Java Networking Fundamentals (Sockets, URLs), TCP/IP Programming, Datagram Sockets (UDP), Remote Method Invocation (RMI), Client-Server Application Development

Unit 4:

Java Database Connectivity (JDBC): Introduction to JDBC API, JDBC Drivers and Architecture, CRUD Operations using JDBC, Connection Pooling, Transaction Management in JDBC,

Unit 5:

Web Programming with Servlets and JSP: Servlet Lifecycle and Architecture, Session Management (Cookies, URL Rewriting, HttpSession), Introduction to Java Server Pages (JSP), JSP Tags, Directives, and Error Handling, MVC Architecture in Java Web Applications.

Textbooks & Reference Books:

1."Core Java Volume II: Advanced Features" by Cay S. Horstmann and Gary Cornell

2."Java: The Complete Reference" by Herbert Schildt

Reference Books:"Head First Servlets and JSP" by Bryan Basham, Kathy Sierra, and Bert Bates

1."Spring in Action" by Craig Walls

2."Hibernate in Action" by Christian Bauer and Gavin King

Advance Java Lab
Course Code:M020224252

Year 1st **L T P C**

COURSEOBJECTIVES
Implement advanced OOP concepts using generics, collections, and lambda expressions.
Develop multi-threaded applications and manage concurrency using the ExecutorService.
Create client-server applications using Java networking APIs (Sockets).
Perform database operations with JDBC and develop web applications using Servlets and JSP.

Semester-2nd **0 0 4 2**

Course Outcome (CO)	Bloom's Knowledge Level (KL)	
At the end of course, the student will be able to understand		
CO 1	Implement generics, collections, and lambda expressions in Java.	K2,K3
CO 2	Develop multi-threaded applications using Thread, Runnable, and ExecutorService.	K3,K4
CO 3	Build client-server programs using Java sockets.	K3,K4
CO 4	Connect to databases and perform CRUD; develop web applications with Servlets and JSP.	K4,K5

List of Experiments/Practical

S.No	Name of Experiment
1.	i)Implement a generic class and methods. ii) Create and perform operations on different collection types (List, Set, Map, Queue) using Generics.
2.	Write a program that demonstrates the use of lambda expressions and functional interfaces.
3.	Create a program that implements threads using both `Thread` class and `Runnable` interface.
4.	Implement a program using the `ExecutorService` to manage threads in a thread pool.
5.	Create a program that uses `java.net.Socket` to establish a client-server connection.
6.	Implement server-side socket programming to listen for client requests.
7.	Establish a JDBC connection to a relational database (e.g., MySQL).
8.	Perform basic CRUD operations (Create, Read, Update, Delete) on a database table using JDBC.
9.	Develop a simple Servlet-based application that demonstrates the request-response lifecycle.
10	Create a basic JSP page to display dynamic content, using tags, directives, and expressions.

Advance Database Management System Lab
(CourseCode:M020224251)

COURSEOBJECTIVES
Understand Database Concepts: Learn the fundamental principles of database design and management. **Practice SQL Queries: Gain proficiency in writing and executing SQL queries for data manipulation and retrieval.** **Database Normalization: Apply normalization techniques to optimize database structures.**
Practice SQL Queries: Gain proficiency in writing and executing SQL queries for data manipulation and retrieval.
Database Normalization: Apply normalization techniques to optimize database structures.

Year:1st **L T P C**

Semester:IInd **0 0 2 1**

CourseOutcome(CO)		**Bloom'sKnowledgeLevel(KL)**
Attheendofcourse,thestudentwill beabletounderstand		
CO1	Apply SQL for data definition and manipulation in real-world scenarios. Design normalized databases to reduce redundancy and improve efficiency.	K3
CO2	Demonstrate the use of DBMS tools for database design and query execution	K6
CO3	Implement and manage database transactions with proper concurrency control	K3
CO4	Implement and manage database transactions with proper concurrency control	K3
CO5	Understand database security and apply mechanisms to safeguard data integrity	K2.K3

1. Installingoracle/MYSQL/Xampp Server
2. CreatingEntity-RelationshipDiagramusingcasetools.
3. Reduce the Entity – Relationship Diagram into the table with proper rules.
4. WritingSQLstatementsUsingORACLE/MYSQL:
    a) Create database and table.
    b) Writing basic SQL SELECT statements.
    c) Restricting and sorting data.
    d) Group by and having clause
    e) Displaying data from multiple tables.
    f) Aggregating data using group function.
    g) Manipulating data.
    h) Managing and updating data and structure of tables.
5. Different operation of database like rename the database.
6. Joins in SQL
7. Normalization
8. Case expression
9. Index of the table
10. Creatingcursor
11. Creatingprocedureandfunctions
12. Creatingpackagesand triggers
13. Subqueries in SQL
14. Designandimplementationofpayrollprocessingsystem
15. Design the database for student management system.
16. Design a database for E-commerce website.
17. AutomaticBackupofFilesandRecoveryofFiles
18. Miniproject(Design&DevelopmentofDataandApplication)forfollowing:
    a) InventoryControl System.
    b) Airline management system.
    c) MaterialRequirement Processing.
    d) HospitalManagementSystem.
    e) RailwayReservation System.
    f) PersonalInformationSystem.
    g) WebBasedUserIdentification System.
    h) TimetableManagementSystem.
    i) HotelManagement System

School Name- School of Computer Applications			
Program- MCA			**Semester-3rd**
Course Name- Web Technology			
A.Y 2025-26	**Course Code- M020224301**	**Batch- 2023- 27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 50	**Total Credits- 4**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks 100 (MM)**

Course Objectives/Course Description

1. To introduce the fundamentals of web development, including web architecture, protocols, HTML, CSS. and responsive design principles.
2. To develop skills in client-side scripting using JavaScript for creating dynamic and interactive web interfaces.
3. To enable the development of server-side applications using Servlets, JSP, and session management techniques.
4. To build full-stack web applications using the Spring framework and Spring Boot, including RESTful web services and modern configuration approaches.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
1	**Web Page Designing:** Introduction and Web Development Strategies, History of Web and Internet, Protocols Governing Web, HTML-Introduction, HTML Tags, HTML-Grouping Using Div& Span, HTML-Lists, HTML-Images, HTML-Hyperlink, HTML-Table, HTML-Iframe, HTML-Form, Introduction of CSS, CSS Syntax, External Style Sheet using < link >, Multiple Style Sheets, Value Lengths and Percentages, CSS-Selectors, CSS-Box Model, Floats, Clear, Introduction to Bootstrap.	**10**
2	**Scripting:** Introduction to JavaScript, Creating Variables in JavaScript, Creating Functions in JavaScript, UI Events, Returning Data from Functions, Working with Conditions, looping in JavaScript, Block Scope Variables, Working with Objects, Creating Object using Object Literals, Manipulating DOM Elements with JavaScript	**10**
3	**Web Application development using JSP & Servlets:** Servlet Overview and Architecture, Interface Servlet and the Servlet Life Cycle, Handling HTTP get Requests, Handling HTTP post Requests, Redirecting Requests to OtherResources, Session Tracking, Cookies, Session Tracking with Http Session. Java Server Pages (JSP): Introduction, Java Server Pages Overview, A First Java Server PageExample,ImplicitObjects,Scripting,StandardActions,Directives,CustomTagLibraries.	**10**
4	**Spring:**Spring Core Basics-Spring Dependency Injection concepts, Introductionto Design patterns, Factory Design Pattern, Strategy Design pattern, Spring Inversion of Control, AOP, Bean Scopes- Singleton, Prototype, Request, Session, Application, WebSocket, Auto wiring, Annotations, Life Cycle Call backs, Bean Configuration styles	**10**
5	**Spring Boot:** Spring Boot- Spring Boot Configuration, Spring Boot Annotations,	**10**

	Spring Boot Actuator, Spring Boot Build Systems, Spring Boot Code Structure, SpringBootRunners,Logger,BUILDINGRESTFULWEBSERVICES,Rest Controller, Request Mapping, Request Body, Path Variable, Request Parameter, GET, POST, PUT, DELETE APIs, Build Web Applications	

Course Outcomes

CO1: ApplytheknowledgeofHTMLandCSStodevelopwebapplicationand analyzetheinsightsofinternetprogrammingtoimplementcomplete application over the web.
CO2: Understand,analyzeandapplytheroleofJavaScriptintheworkingsofthe web and web applications.
CO3: Understand,analyzeandbuilddynamicwebapplicationsusingservletandJSP
CO4: DevelopSpring-basedJavaapplicationsusingJavaconfiguration,XML configuration,annotation-basedconfiguration,beansandtheirscopes,and properties.
CO5: DevelopwebapplicationusingSpringBootandRESTFulWebServices.

Text books:

1. Burdman,Jessica,"CollaborativeWebDevelopment"AddisonWesley
2. Xavier,C,"WebTechnologyandDesign",NewAgeInternational
3. IvanBayross,"HTML,DHTML,JavaScript,Perl&CGI",BPBPublication
4. Bhave,"ProgrammingwithJava",PearsonEducation

Reference books:

1. HansBergsten,"JavaServerPages",SPDO'Reilly
2. Naughton,Schildt,"TheCompleteReferenceJAVA2",TMH
3. CraigWalls,"SpringBootinAction"

Assessment method: (Continuous Internal Assessment = 50th%, Final Examination = 50th%)

School Name- School of Computer Applications			
Program- MCA			**Semester-3rd**
Course Name- Web Technology Lab			
A.Y 2025-26	**Course Code- M020224351**	**Batch- 2024-26**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15	**Total Credits- 1**		**ESE Marks- 40 (MM)**
Type of Course- practical			**Total Marks 100 (MM)**

Course Objectives/Course Description

- Design static and dynamic web pages using HTML, CSS, and JavaScript.
- Create interactive front-end components with client-side scripting.
- Implement basic backend logic using server-side scripting languages.
- Connect web applications with databases.
- Deploy a full-stack mini web project

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
	1. **Design a static personal homepage** using HTML5. 2. **Create a form** with various input types (text, radio, checkbox, select, etc.). 3. **Use CSS3 to style** a web page: layout, fonts, colors, margins, and paddings. 4. **Implement a responsive webpage** using media queries or Bootstrap. 5. **Form validation using JavaScript** (e.g., email, password length). 6. **Use JavaScript to manipulate the DOM:** show/hide elements, change styles, or create sliders. 7. **Create a basic calculator** or to-do list using JavaScript. 8. **Introduction to PHP or Node.js or Python (Flask/Django):** • Display current time/date • Read and display form data using POST/GET methods 9. **User login page:** • Simple authentication using hardcoded credentials or form input 10. **Connect a web form to a database** (e.g., MySQL or SQLite): • Store user input • Retrieve and display records 11. **Create a basic CRUD application:** • Add, view, edit, delete operations • Example: student records or product catalog 12. **Develop a mini web application** combining front-end and back-end:Example Projects: • Online feedback form • Bookstore / product catalog • Blog or news portal • Student registration system	

Course Outcomes

CO1: Design and develop static and responsive web pages using HTML5, CSS3, and Bootstrap for

cross-device compatibility.
CO2: Create interactive web forms with various input types and implement client-side form validation using JavaScript.
CO3: Manipulate the Document Object Model (DOM) using JavaScript to enhance web page interactivity and functionality.
CO4: Develop basic dynamic web applications using server-side scripting languages (PHP, Node.js, or Python frameworks) to handle user inputs and display data.
CO5: Build database-connected CRUD applications to store, retrieve, update, and delete data, integrating front-end and back-end components into mini web projects.

Assessment method: (Continuous Internal Assessment = 60th%, Final Examination = 40th%)

School Name- School of Computer Applications			
Program- MCA			**Semester-3rd**
Course Name- DeepLearning			
A.Y 2025-26	**Course Code- M020224304**	**Batch- 2023-27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 50	**Total Credits- 4**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks 100 (MM)**

Course Objectives/Course Description

1. To provide foundational knowledge of neural networks and deep learning, including key mathematical and computational principles.
2. To explore deep learning architectures and techniques, such as convolutional neural networks, auto encoders, RNNs, LSTMs, and GANs.
3. To introduce dimensionality reduction methods and optimization strategies essential for building efficient and generalizable deep learning models.
4. To apply deep learning techniques to real-world problems through case studies in computer vision, natural language processing, and other domains.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
1	INTRODUCTION:Introductiontomachinelearning-Linearmodels(SVMsandPerceptrons, logisticregression)-IntrotoNeuralNets:Whatashallownetworkcomputes-Traininganetwork: lossfunctions,backpropagationandstochasticgradientdescent-Neuralnetworksasuniversal function approximates	**10**
2	DEEP NETWORKS : History of Deep Learning- A Probabilistic Theory of Deep Learning-Backpropagationandregularization,batchnormalization-VCDimensionandNeuralNets-DeepVs ShallowNetworks-ConvolutionalNetworks-GenerativeAdversarialNetworks(GAN),Semi-supervisedLearning	**10**
3	DIMENTIONALITY REDUCTION 9 Linear (PCA, LDA) and manifolds, metric learning - Auto encoders and dimensionality reduction in networks - Introduction to Convnet - Architectures – AlexNet, VGG, Inception, ResNet - Training a Convnet: weights initialization, batchnormalization, hyper parameter optimization	**10**
4	OPTIMIZATION AND GENERALIZATION : Optimization in deep learning– Non-convex optimizationfordeepnetworks-StochasticOptimizationGeneralizationinneuralnetworks-Spatial Transformer Networks- Recurrent networks, LSTM - Recurrent Neural Network LanguageModels-Word-LevelRNNs&DeepReinforcementLearning-Computational&ArtificialNeuroscience	**10**
5	CASE STUDY AND APPLICATIONS : Image net- Detection-Audio Wave Net-Natural Language Processing Word2Vec - Joint Detection-Bioinformatics- Face Recognition- Scene Understanding- Gathering	**10**

	Image Captions	

Course Outcomes

CO1: Topresentthemathematical,statisticalandcomputationalchallengesofbuildingneuralnetworks
CO2: Tostudytheconceptsofdeeplearning
CO3: Tointroducedimensionalityreductiontechniques
CO4: Toenablethestudentstoknowdeeplearningtechniquestosupportreal-timeapplications
CO5: Toexaminethecasestudiesofdeeplearningtechniques.

Text books:

1. CosmaRohillaShalizi,AdvancedDataAnalysisfromanElementaryPointofView,2015.
2. Deng&Yu,DeepLearning:MethodsandApplications,NowPublishers,2013.
3. IanGoodfellow,YoshuaBengio,AaronCourville,DeepLearning,MITPress,2016.

MichaelNielsen,NeuralNetworksandDeepLearning,DeterminationPress,2015.

Reference books:

1. IanGoodfellow,YoshuaBengio,AaronCourville,DeepLearning,MITPre ss,2016.
2. MichaelNielsen,NeuralNetworksandDeepLearning,DeterminationPres s,2015.

Assessment method: (Continuous Internal Assessment = 50th%, Final Examination = 50th%)

School Name- School of Computer Applications			
Program- MCA			**Semester-3rd**
Course Name- Deep Learning Lab			
A.Y 2025-26	**Course Code- M020224354**	**Batch- 2023-27**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15	**Total Credits- 1**		**ESE Marks- 40 (MM)**
Type of Course- Practical			**Total Marks 100 (MM)**

Course Objectives/Course Description

- Implement deep learning models using Python and deep learning libraries.
- Train, validate, and test neural networks on real-world datasets.
- Understand how deep learning models work internally (layers, activation, optimization).
- Use convolutional neural networks (CNNs) and recurrent neural networks (RNNs) for image and sequence data.
- Evaluate models using appropriate metrics and tuning techniques.

UNIT-	Topics	No. of Teaching hours/ (Lecture)
	1. **Introduction to NumPy and TensorFlow/Keras** o Matrix operations, broadcasting, activation functions 2. **Build a single-layer perceptron model** o For basic classification tasks (e.g., AND/OR logic gates) 3. **Implement a simple multi-layer perceptron (MLP)** o For handwritten digit recognition using MNIST dataset 4. **Explore activation functions** o ReLU, Sigmoid, Tanh: Compare their performance 5. **Model evaluation metrics**Accuracy, precision, recall, F1-score 6. **Build a basic CNN model using Keras** o For image classification (e.g., CIFAR-10 or MNIST) 7. **Experiment with data augmentation and dropout** o Improve model generalization and reduce overfitting 8. **Build a simple RNN or LSTM** o For sequence prediction (e.g., next word or character prediction) 9. **Use preprocessed text datasets** o Sentiment analysis using LSTM/GRU models 10. **Transfer learning using pre-trained models** o Use VGG16 or MobileNet for custom image classification 11. **Hyperparameter tuning and callbacks** o Using validation split, learning rate schedulers, and early stopping	

Course Outcomes

CO1: Apply foundational concepts of deep learning using libraries like NumPy, TensorFlow, and Keras, including matrix operations and activation functions.

CO2: Design and implement neural network models such as single-layer and multi-layer perceptrons

(MLPs) for classification tasks using datasets like MNIST.
CO3: Build, train, and evaluate advanced deep learning models such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs/LSTMs) for image and sequence data.
CO4: Utilize techniques such as data augmentation, dropout, and transfer learning with pre-trained models (e.g., VGG16, MobileNet) to enhance model performance.
CO5: Analyze model performance using metrics like accuracy, precision, recall, and F1-score, and apply hyperparameter tuning and callbacks (e.g., early stopping, learning rate schedulers) to optimize training.

Assessment method: (Continuous Internal Assessment = 60th%, Final Examination = 40th%)

School Name- School of Computer Applications			
Program- MCA			**Semester-3rd**
Course Name- Advance Linear Algebra Probability and Statistics for Machine Learning			
A.Y 2025-26	**Course Code- M020224303**	**Batch- 2023-27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 50	**Total Credits- 4**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks 100 (MM)**

Course Objectives/Course Description

- Understand and apply core concepts of linear algebra including matrices, vectors, and eigen systems.
- Develop foundational knowledge of probability theory and distributions relevant to real-world problems.
- Apply statistical techniques to analyze and interpret data effectively.
- Use computational tools to perform mathematical operations, data analysis, and simulations.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
1	**Introduction to Linear Algebra** • Scalars, Vectors, Matrices, and Tensors: Definitions and examples • Matrix operations: Addition, multiplication, transpose, inverse • Special matrices: Identity, Diagonal, Symmetric, Orthogonal • Determinant and Rank of a matrix	**10**
2	**Systems of Linear Equations and Eigen Concepts** • Solving linear systems via Gaussian elimination • Eigenvalues and Eigenvectors: Concepts, computation, and applications • Singular Value Decomposition (SVD): Basic understanding • Vector spaces, subspaces, linear independence	**10**
3	**Fundamentals of Probability** • Basic probability theory: Sample space, events, axioms • Conditional probability and Bayes' theorem • Independence of events • Random variables: Discrete and continuous • Expectation, variance, and standard deviation	**10**
4	**Probability Distributions and Relationships** • Probability distributions: Binomial, Poisson, Uniform, Normal (Gaussian) • Joint, marginal, and conditional distributions • Visualizing and interpreting probability models	**10**
5	**Statistics and Data Inference** • Descriptive statistics: Mean, median, mode, range, variance,	**10**

	standard deviation • Measures of central tendency and dispersion • Correlation and covariance • Introduction to hypothesis testing and confidence intervals • Sampling methods and simple linear regression (least squares method)	

Course Outcomes

CO1: Perform matrix operations and solve systems of linear equations using analytical and computational methods.

CO2: Calculate and interpret eigenvalues, eigenvectors, and apply basic concepts of vector spaces and SVD.

CO3: Apply basic and conditional probability, Bayes' theorem, and simulate random variables using programming tools.

CO4: Understand and work with various probability distributions and analyze their properties.

CO5: Perform descriptive statistics, correlation analysis, hypothesis testing, and simple linear regression on datasets using software tools like Python or R.

Text books:

1. Linear Algebra and Its Applications by Gilbert Strang
2. Probability and Statistics for Engineers and Scientists by Ronald Walpole

Reference books:

1. Introduction to Probability and Statistics by William Mendenhall
2. Think Stats by Allen B. Downey (free online)

Assessment method: (Continuous Internal Assessment = 50th%, Final Examination = 50th%)

<table>
<tr><td colspan="4">School Name- School of Computer Applications</td></tr>
<tr><td colspan="3">Program- MCA</td><td>Semester-3rd</td></tr>
<tr><td colspan="4">Course Name- Project</td></tr>
<tr><td>A.Y 2025-26</td><td>Course Code- M020224352</td><td>Batch- 2023-27</td><td>CIE Marks- 60 (MM)</td></tr>
<tr><td>Total Teaching Hours 50</td><td colspan="2">Total Credits- 4</td><td>ESE Marks- 40 (MM)</td></tr>
<tr><td colspan="3">Type of Course- Theory</td><td>Total Marks 100 (MM)</td></tr>
<tr><td colspan="4">Course Objectives/Course Description</td></tr>
<tr><td colspan="4">Course Outcomes
CO 1-Developing a technical artifact requiring new technical skills and effectively utilizing a new software tool to complete a task
CO 2- Writing requirements documentation, Selecting appropriate technologies, identifying and creating appropriate test cases for systems.
CO 3-Demonstrating understanding of professional customs & practices and working with professional standards.
CO 4- Improving problem-solving, critical thinking skills and report writing.
CO 5-Learning professional skills like exercising leadership, behaving professionally, behaving ethically, listening effectively, participating as a member of a team, developing appropriate workplace attitudes</td></tr>
</table>

Assessment method: (Continuous Internal Assessment = 60th%, Final Examination = 40th%)

SDGI GLOBAL UNIVERSITY (SGU)

Syllabus- Research Writing Skills (Subject Code: SGUAE2405)

Unit-1: Introduction to Research Writing:

- The importance and Purpose of Research Writing
- Types of Research Papers
- Understanding the Research Process
- Identifying Research Problems and Formulating Research Questions
- Developing Testable Hypotheses

Unit-2: Literature Review and Research Methodologies:

- Techniques for Conducting Literature Searches
- Evaluating Sources for Credibility
- Qualitative, Quantitative and Rhetorical Research Methods
- Experimental and Non-Experimental Research Designs
- Data Collection and Data-Analysis Techniques
- Thesis, Synopsis Writing-Structure and Importance

Unit-3: Structuring and Writing Research Papers:

- Components of Research Paper: Title Page, Abstract, Introduction, Literature Review, Methodology, Results, Discussion, Conclusion, Citing Sources etc.
- Writing the Methodology Section: Detailing Research Methods and Procedures
- Presenting Results: Using tables, Figures and Descriptive Statistics
- Writing the Discussion: Interpreting Findings and Discussing Implications

Unit-4: Ethical Considerations and Citations:

- Avoiding Plagiarism and Academic Dishonesty
- Ethical Issues in Research and Publications
- Various Citation Styles: APA, MLA, Chicago etc.
- Proper Formatting and Referencing

Unit-5: Effective Research Presentations:

- Introduction, Purpose, Personal Skills, Language Skills, Content Development, Gathering Supporting Evidence
- Techniques for Effective Revision and Editing
- Incorporating Feedback and Final Proofreading
- Preparing and Delivering a Research Paper Presentation

SDGI Global University

Value Added Course

Subject: Human Values & Professional Ethics	Type: Theory	L	T	P	C
		3	0	0	3
Total Hours:45	Sub. Code: BSGUVA2402	Maximum Marks:100			

COURSE OBJECTIVE

To help students distinguish between values and skills, and understand the need, basic guidelines, and process of value education. To help students initiate a process of dialog within themselves to know what they 'really want to be in their life. To help students understand the meaning of happiness and prosperity for a human being Human living, and living accordingly.

Expected Course Outcomes

Course Outcome No.	At the end of the course, the learner will be able to do the following.	Cognitive level
CO1	Understand the significance of value inputs in a classroom, distinguish betweenvaluesandskills,understandtheneed,basicguidelines,contentand process of value education,explorethe meaningofhappiness and prosperity and do a correct appraisal of the current scenario in the society	K 1
CO2	Distinguish between the Self and the Body, understand the meaning of Harmony in the Self the Co-existence of Self and Body	K 2
CO3	Understand the value of harmonious relationship based on trust, respect and other naturally acceptable feelings in human-human relationships and explore their role in ensuring a harmonious society	K3
CO4	Understand the harmony in nature and existence, and work out their mutually fulfilling participation in the nature.	K4
CO5	Tofacilitatethestudentsinapplyingtheunderstandingofharmonyin existence in their profession and lead an ethical life	K 5

Detailed Syllabus

UNIT 1: Course Introduction - Need, Basic Guidelines, Content and Process for Value Education (9 hour)

1. Understandingthe need and basic guidelinesforValueEducation
2. Self-Exploration–whatisit?-Itscontentandprocess; on the basis of NaturalAcceptance
3. Rightunderstanding,RelationshipandPhysicalFacilities-thebasicrequirementsforfulfilmentofaspirationsofeveryhumanbeingwiththeir correct priority
4. UnderstandingHappinessandProsperitycorrectly-Acriticalappraisalofthecurrent scenario atvariouslevel

UNIT 2: Understanding Harmonyin theHumanBeing-HarmonyinMyself! (9Hours)

1. Understandinghumanbeingasaco-existenceofthesentient‘I’andthematerial‘Body’
2. UnderstandingtheneedsofSelf(‘I’)and ‘Body’-*SukhandSuvidha*
3. UnderstandingtheBodyas an instrumentof ‘I’ (Ibeingthe doer, seerandenjoyer)
4. Understandingtheactivities in the selfand activities in the body.
5. HarmonywiththeBody:*Sanyam*and*Swasthya*(Correct appraisalofPhysicalneeds)

UNIT3:UnderstandingHarmonyintheFamilyandSociety-HarmonyinHuman-HumanRelationship. (9Hours)

1. UnderstandingHarmonyinthefamily–thebasicunitofhumaninteraction
2. Four element of Justice; Ensure Mutual Happiness
3. Values in Human Relationship (Nine Values): Understanding the meaning of Values
4. Foundation Value (Trust): Difference between intention and competence
5. UndividedSociety(*AkhandSamaj*) From Individual toworldfamily.

UNIT 4:UnderstandingHarmonyintheNatureandExistence-Wholeexistenceas Co-existence. . (9 Hours)

1. Understanding theharmonyintheNature: The Four Order in Nature
2. Interconnectednessandmutualfulfilmentamongthefourordersofnature-recyclabilityandself-regulationinnature.
3. UnderstandingExistenceasCo-existence*(Sah-astitva*)ofmutuallyinteracting
4. Holisticperception of harmonyat alllevelsof existence.

UNIT 5: Implication of the above Holistic Understanding of Harmony onProfessionalEthics (9 Hours)

1. Basis for Universal Human Values and Ethical Human Conduct
2. Professional Ethics in Light of Right Understanding.
3. Vision for Holistic Technologies, Production Systems and Management Models
4. Evolving Holistic Models of Living

Books reference: -

1.R R Gaur, R Sangal, G P Bagaria, 2009, A Foundation Course in Human Values and Professional Ethics.
2.E G Seebauer& Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers , Oxford University Press
3.M Govindrajran, S Natrajan& V.S. Senthil Kumar, Engineering Ethics (including Human Values), Eastern Economy Edition, Prentice Hall of India Ltd.

Text Books

1. RRGaur,RSangal,GPBagaria,2009,AFoundationCourseinHuman Values and Professional Ethics.

Reference Book

Ivan Illich, 1974, Energy & Equity, The Trinity Press, Worcester, and Harper Collins, USA
E.F. Schumacher, 1973, small is Beautiful: a study of economics as if people mattered, Blond &Briggs, Britain.
Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986, 1991
Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, Limits to Growth – Club of Rome’s report, Universe Books.

A Nagraj, 1998, JeevanVidyaEkParichay, Divya Path Sansthan, Amarkantak.