Centre for Distance and Online Education
Central University of Himachal Pradesh
Dharamshala (H.P.)- INDIA

Programme Structure and Detailed Syllabus

Master of Computer Applications (MCA)

TWO YEAR ONLINE PROGRAMME

Centre for Distance and Online Education (CDOE)
Central University of Himachal Pradesh
Dharamshala, Distt. Kangra (HP) – INDIA

Programme Structure of
Two Year Online Master of Computer Applications (MCA) Programme

Master of Computer Applications (MCA) Online 1st Semester			
Course Code	**Name of the Course**		**Name of Course Coordinator**
MCA-501	Data Structures	4	Mr. Ajay Kumar
MCA-502	Data Structures-Lab	2	Mr. Ajay Kumar
MCA-503	Operating System	4	Dr. Parveen Sadotra
MCA-521	Software Engineering	2	Dr. Pankaj Kumar
MCA-534	Object Oriented Programming	4	Mr. Manoj Dhiman
MCA IKS	Indian Knowledge System	2	Dr. B.C. Chouhan
MGT-528	Business Communication Skills	2	Dr. Aditi Sharma
MCA-403	Fundamentals of Computer & Programming in C **(Bridge Course)**	4	Mr. Mayank Chopra
Master of Computer Applications (MCA) Online 2nd Semester			
Course Code	**Name of the Course**		**Name of Course Coordinator**
MCA-504	Theory of Computation	4	Mr. Mayank Chopra
MCA-505	Database Management System	4	Mr. Manoj Dhiman
MCA-506	Database Management System Lab	2	Mr. Manoj Dhiman
MCA-507	Design & Analysis of algorithms	2	Mr. Ajay Kumar
MCA-522	Software Testing	2	Dr. Pankaj Kumar
MCA-527	Python Programming	2	Mr. Mayank Chopra
MCA-551	IT Tools for IKS	2	Dr. Pradeep Chouksey
HIS-473	Cultural Heritage of Bharat	2	Dr. Pravat Ranjan Sethi
Master of Computer Applications(MCA) Online 3rd Semester			
MCA-533	Cyber Security	4	Dr. Parveen Sadotra
MCA-	Research Methodology	4	Dr. Pradeep Chouksey

601			
MCA-516	Web Programming	4	Mr. Mayank Chopra
MCA-604	Literature Review	4	Mr. Mayank Chopra
MCA-605	Research Proposal	4	Mr. Manoj Dhiman
Master of Computer Applications(MCA) Online 4th Semester			
MCA-519	Computer Networks	4	Dr. Parveen Sadotra
MCA-602	Academic writing	2	Dr. Parveen Sadotra
MCA-603	Research Paper Publication	2	Dr. Pradeep Chouksey
MCA-536	Java Programming	4	Mr. Ajay Kumar
MCA-606	Dissertation	8	Dr. Parveen Sadotra

The course-wise detailed syllabus is provided as follows:

SEMESTER I

Detailed Syllabi of the Courses for All Four Semesters of Two Year Online Master of Computer Applications MCA Programme

Course Code: MCA 501

Name of Course: Data Structure

Course Coordinator: Mr. Ajay Kumar

Credit: 4

Course Objectives:

- To equip students with a comprehensive understanding of core data structures (arrays, linked lists, stacks, queues, trees, graphs, hash tables) and their underlying principles.
- To cultivate student's ability to analyze problem requirements, design efficient algorithms using suitable data structures, and evaluate the performance (time and space complexity) of their solutions.
- To empower students to write optimized and scalable code by mastering the practical implementation of data structures, leading to the creation of robust and efficient software applications.
- To train students in the effective storage, retrieval, and manipulation of data using various data structures.

Course Outcomes: After the successful completion of this course, the student will be able to

- Learn the basics of data structures such as arrays, linked lists, stacks, queues, trees, and graphs to efficiently store and manage data.
- Develop skills to implement data structures that optimize data retrieval, insertion, deletion, and searching operations for better performance in software applications.
- Gain expertise in designing and analyzing algorithms using appropriate data structures to improve computational efficiency and solve complex problems effectively.
- Apply data structure techniques in software development, databases, artificial intelligence, networking, and other computing domains to build scalable and efficient applications.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content:

Unit-I: Introduction

Module/ Chapter 1: Introduction to Data Structures

Definition, classification, need for data structures, examples of time and space complexity, asymptotic notations (Big O, Omega, Theta)

Module/ Chapter 2: Array Manipulations

Definition, representation, dynamic memory allocation, single and two-dimensional arrays, sparse matrices, address calculation, traversing, insertion & deletion in arrays

Module/ Chapter 3: Searching and Sorting Algorithms

Linear Search, Binary Search, Insertion Sort, Bubble Sort, Selection Sort, Counting Sort, Radix Sort, Time complexity analysis of sorting algorithms

Unit-II: Linked List

Module/ Chapter 4: Introduction to Linked Lists

Definition, concepts, types of linked lists (Singly, Doubly, Circular), comparison: Array vs Linked List, memory allocation (Garbage Collection), traversing & searching in linked lists

Module/ Chapter 5: Singly Linked List

Basic operations (creation, insertion, deletion, traversal), insertion at beginning, end, and specific locations, deletion from beginning, end, and specific locations

Module/ Chapter 6: Doubly and Circular Linked Lists

Basic operations on doubly linked list (creation, insertion, deletion, traversal), circular linked list concepts, advantages over singly linked lists, real-world applications.

Unit-III: Stack and Queue

Module/ Chapter 7: Stack: Concept and Implementation

Definition, LIFO principle, operations (push, pop, peek), array implementation of stack, linked list implementation of stack

Module/ Chapter 8: Applications of Stack

Expression conversion (Infix to Postfix, Infix to Prefix), expression evaluation, recursion using stack, backtracking applications (maze solving, N-Queens problem)

Module/ Chapter 9: **Queue and Its Variants**

Definition, FIFO principle, operations (enqueue, dequeue, peek), array and linked list implementation, types of queues (simple queue, circular queue, priority queue, double-ended queue - Deque

Unit-IV: Trees

Module/ Chapter 10: **Introduction to Trees**

Tree terminology (nodes, root, parent-child, siblings, depth, height), properties of binary trees, types of trees (general tree, binary tree, binary search tree, AVL tree, B-tree, B+ tree), advantages of tree structures

Module/ Chapter 11: **Binary Tree and Its Traversal Techniques**

Array and linked representation of binary trees, tree traversal techniques (Preorder, Inorder, Postorder, Level-order traversal), expression trees

Module/ Chapter 12: **Binary Search Tree (BST) and Its Applications**

Definition, properties, basic operations (insertion, deletion, searching), self-balancing BSTs (AVL Tree, Red-Black Tree), applications of BST

Unit-V: Advanced Topics

Module/ Chapter 13: **Heap and Hashing**

Introduction to heaps (Min-Heap, Max-Heap), heap operations (insertion, deletion), applications in priority queues, hashing concepts (hash functions, collision handling techniques - chaining, open addressing)

Module/Chapter 14: **Graph Data Structure and Applications**

Definition, types (directed, undirected, weighted, unweighted), graph representation (adjacency matrix, adjacency list), graph traversal algorithms (DFS, BFS), applications in networking, AI, and route optimization

Suggested Readings:

1. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
2. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.
3. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using C and C++:, Second edition, PHI, 2009.
4. Robert L. Kruse, "Data Structures and Program Design in C++", Pearson.
5. D.S Malik, Data Structure using C++,Second edition, Cengage Learning, 2010.

Course Code: MCA 502

Name of Course: Data Structure Lab

Course Coordinator: Mr. Ajay Kumar

Credit: 2

Course Objectives:

- To equip students with a comprehensive understanding of core data structures (arrays, linked lists, stacks, queues, trees, graphs, hash tables) and their underlying principles.
- To cultivate student's ability to analyze problem requirements, design efficient algorithms using suitable data structures, and evaluate the performance (time and space complexity) of their solutions.
- To empower students to write optimized and scalable code by mastering the practical implementation of data structures, leading to the creation of robust and efficient software applications.
- To train students in the effective storage, retrieval, and manipulation of data using various data structures.

Course Outcomes: After the successful completion of this course, the student will be able to

- Learn the basics of data structures such as arrays, linked lists, stacks, queues, trees, and graphs to efficiently store and manage data.
- Develop skills to implement data structures that optimize data retrieval, insertion, deletion, and searching operations for better performance in software applications.
- Gain expertise in designing and analyzing algorithms using appropriate data structures to improve computational efficiency and solve complex problems effectively.
- Apply data structure techniques in software development, databases, artificial intelligence, networking, and other computing domains to build scalable and efficient applications.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content:

Unit-I: Array data structure

Module/ Chapter 1:	**Basic Array operations**
(Programming List)	Insert an element at a given position in an array, Delete an element from a given position in an array, Print all

elements of an array, Find an element in an unsorted array, Find an element in a sorted array

Module/ Chapter 2: **Sorting Algorithms**

(Programming List) Sort an array using the bubble, selection, insertion, counting, and radix, sort algorithm

Unit-II: Linked List data structure

Module/ Chapter 3: **Basic Linked List operations**

(Programming List) Implement a singly linked list and insert elements, Print all elements of a linked list, Count the number of nodes in a linked list, Find if an element exists in a linked list, reverse a linked list

Module/ Chapter 4: **Singly Linked List**

(Programming List) Implement Singly Linked List operations: insertion at beginning, end, and specific locations, deletion from beginning, end, and specific locations

Module/ Chapter 5: **Doubly Linked Lists**

(Programming List) Implement Doubly Linked List operations: insertion at beginning, end, and specific locations, deletion from beginning, end, and specific locations

Unit-III: Stack data structure

Module/ Chapter 6: **Stack Implementation using Arrays**

(Programming List) Implement a stack using a fixed-size array, Implement push, pop, peek, isEmpty, and isFull operations

Module/ Chapter 7: **Stack Implementation using Linked Lists**

(Programming List) Implement a stack using a singly linked list, Implement push, pop, peek, and isEmpty operations

Module/ Chapter 8: **Applications of Stack**

(Programming List) Use a stack to reverse a given string, check if a string is a palindrome, check if a given string of parentheses is balanced, Implement the Tower of Hanoi problem using recursion

Unit-IV: Queue data structure

Module/ Chapter 9: **Queue Implementation using Arrays**

(Programming List) Implement a queue using a fixed-size array, Implement enqueue, dequeue, peek, isEmpty, and isFull operations

Module/ Chapter 10: Queue Implementation using Linked List

(Programming List) Implement a queue using a singly linked list, Implement enqueue, dequeue, peek, and isEmpty operations.

Suggested Readings:

6. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
7. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.
8. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using C and C++:, Second edition, PHI, 2009.
9. Robert L. Kruse, "Data Structures and Program Design in C++", Pearson.
10. D.S Malik, Data Structure using C++,Second edition, Cengage Learning, 2010.

Course Code: MCA 503

Name of Course: **Operating System**

Course Coordinator: **Dr. Parveen Sadotra**

Credit: **4**

Course Objectives:

- ✓ To introduce fundamental concepts, functions, and structures of operating systems.
- ✓ To understand process management, CPU scheduling, memory management, and virtual memory techniques.
- ✓ To explore file systems, I/O management, and process synchronization mechanisms.
- ✓ To analyze deadlocks, their causes, and strategies for prevention, detection, and recovery.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1 - Explain operating system functions, architectures, and process management techniques.

CO^2 - Analyze CPU scheduling, memory management, and page replacement strategies.

CO^3 - Demonstrate knowledge of file system structures, disk scheduling, and synchronization techniques.

CO^4 - Identify and apply methods for deadlock detection, prevention, and recovery.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content

Unit-I: Operating System Introduction

Module/ Chapter 1: Operating System Introduction

Function, Characteristics, Structures/types; simple batch, multi-programmed batch, timesharing

Module/ Chapter 2: Computer System Architectures

Operating-System Structure, Personal computer, parallel system, distributed systems, real-time systems, OS Concepts; system components, operating system services, system calls, virtual machines.

Module/ Chapter 3: Processess

Process concepts and scheduling, Process Management in OS, Attributes of a Process, Process States, Process Schedulers, Process Queues, Times Related to Process

Module/ Chapter 4: CPU Scheduling
Basic Concepts, Operation on Processes, Cooperating processes, threads and inter-process communication scheduling criteria, Scheduling Algorithms; FCFS, SJF, Priority, Round Robin, multiple-processor scheduling, real time scheduling

Unit-II: Memory Management and Virtual memory

Module/ Chapter 5: Memory Management
Logical versus physical address space, swapping contiguous allocation

Module/ Chapter 6: Paging
Paging, segmentation, segmentation with paging, Demand paging, performance of demand paging, page replacement

Module/ Chapter 7: Page Replacement algorithms
FIFO, LRU, OPT,LFU, allocation of frames, thrashing.

Unit-III: File System Interface and Implementation

Module/ Chapter 8: File System-I
Access method, directory, structure, protection, file system structure, allocation methods

Module/ Chapter 9: File System-II
free space's management, directory management, directory implementation, efficiency and performance

Module/ Chapter 10: I/O Management
I/O Software and its types, Disk Scheduling; FCFS, SCAN, C-SCAN, SSTF

Module/ Chapter 11: Process Synchronization
Critical section problem, synchronization, critical regions, monitors

Unit-IV: Deadlocks

Module/ Chapter 12: Introduction of Deadlocks
System model, deadlocks characterization

Module/ Chapter 13: Deadlock handling
Methods for handling deadlocks, deadlock prevention, deadlock avoidance

Module/ Chapter 14: Deadlock detection
deadlock detection, recovery from deadlock

Text Books:

1. Silberschart, Galvin, Gagne, “Operating System Concepts”, Ninth Edition, WSE Wiley.
2. Das, S., Your UNIX: The Ultimate Guide, Fourth Edition, McGraw-Hill Inc.

Reference Book:

1. D.M. Dhamdhere, “Operating Systems: A Concept Based Approach”, Tata McGraw-Hill.
2. Milan Milenkovic, “Operating system-concepts and design”, McGraw Hill International Edition
3. A. S. Godbole, “Operating systems”, Tata McGraw hill
4. Deitel H. M., “Operating System”, Pearson Publications
5. William Stallings, “Operating Systems: Internals and Design Principles”, Prentice-Hall of India
6. Andrew. S. Tanenbaum, “Modern operating systems”, Pearson Prentice Hall.

Course Code: **MCA 521**

Name of Course: **Software Engineering**

Course Coordinator: **Dr. Pankaj Kumar**

Credit: 02

Course Objectives:

- To study fundamental concepts in software engineering concepts, methodologies, and life cycle models.
- To Analyze, document, and manage software requirements effectively.
- To Design software systems using architectural patterns, technique and best practices.
- To Fundamentals of testing techniques to ensure software reliability.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1 - make understanding of software engineering fundamentals, methodologies, and life cycle models.

CO^2 - Elicit, analyze, and document software requirements using appropriate techniques and tools.

CO^3 - Create well-structured software architectures and design models.

CO^4 **-** Conduct unit, integration, and system testing to ensure software reliability and quality.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70%weightage
- Continuous Internal Assessment: 30% weightage

Course Content

UNIT - I: Introduction to Software Engineering

Module/ Chapter 1: Introduction to Software Engineering

The Nature of Software, Definition of Software, Software as a Product and Service, Software Application Domains, Types of Software, Changing Nature of Software

Module/ Chapter 2: Software Engineering Principles and Metrics

The Software Process, The Process Framework, Umbrella Activities, Process Adaptation, Software Engineering Practice,

The Essence of Practice, Definition and Importance of Metrics, Types of Software Metrics, Goals of Metrics

UNIT - II: Software Development Life Cycle and Process Models

Module/ Chapter 3: Overview of Software Development Life Cycle (SDLC)

Definition and Importance of SDLC, Phases of SDLC, Real-world Examples of SDLC Applications

Module/ Chapter 4: Traditional Software Process Models

Characteristics, Advantages, Disadvantages and Applications of Traditional Software Process Models such as Waterfall Model, Prototyping Model, Incremental Model, Comparison between the mentioned models

Module/ Chapter 5: Evolutionary Software Process Models

Characteristics, Advantages, Disadvantages and Applications of Evolutionary Software Process Models such as Iterative Enhancement Model, Spiral Model, Comparison between the mentioned models

UNIT - III: Software Requirement Specification (SRS)

Module/ Chapter 6: Introduction to Software Requirement Specification (SRS)

Definition and Importance of SRS, Role and benefits of SRS in SDLC, Challenges and Pitfalls in SRS Documentation, Characteristics of a Good SRS.

Module/ Chapter 7 Requirements Engineering Process

Functional and non-functional requirements, the software requirements document, Requirements specification, Requirements engineering processes, Requirements elicitation and analysis

Module/ Chapter 8 Requirements Validation and Documentation

Requirements validation, Requirements management

UNIT - IV: System Design and Basics of Testing

Module/ Chapter 9 Introduction to System Design

Introduction to System Design, System Design vs. System Architecture, System Design Process

Module/ Chapter 10 Software Design Methodologies and Testing Basics

Fundamentals of testing, White-box, and black-box testing, Test coverage analysis and test case design techniques, mutation testing.

Text Books:

I. Roger S Pressman Software Engineering: A Practitioner's Approach, 9 edition, McGraw-Hill Higher Education
II. Pankaj Jalote A Concise Introduction to Software Engineering, Springer , 2nd edition

Reference Books:

I. Ian Sommerville Software Engineering, 10/e Pearson Education
II. K.K. Aggarwal Software Engineering, 3rd edition NEW AGE publisher

Course Code	**:**	**MCA 534**
Course Title	**:**	**Object-Oriented Programming**
Course Coordinator	**:**	**Mr. Manoj Dhiman**
Credit	**:**	**4**

Course Objectives:

- To introduce students to the fundamental principles of object-oriented programming (OOP) and its advantages over procedural programming.
- To develop the ability to design and implement efficient, reusable, and maintainable software solutions using C++.
- To apply core OOP concepts such as encapsulation, inheritance, and polymorphism to solve real-world programming problems.
- To explore advanced topics like exception handling, file handling, and the Standard Template Library (STL) for enhanced software development.
-

Course Outcomes: After successful completion of this course, students will be able to:

- **CO[1]** - Explain the fundamental principles of object-oriented programming and its benefits in software development.
- **CO[2]** - Implement OOP concepts such as classes, objects, constructors, inheritance, polymorphism, and operator overloading in C++.
- **CO[3]** - Apply exception handling and file management techniques to ensure robust and error-free application development.
- **CO[4]** - Utilize the Standard Template Library (STL) to develop efficient and scalable applications.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Unit-I: Fundamentals of Object-Oriented Programming

Module/ Chapter 1: Introduction to C++ Programming
Overview of C++, Basic Syntax and Structure, Input and Output, Operators and Expressions, Type Conversion.

Module/ Chapter 2: Control Structures
Control Structures: Decision-making, Looping, and Jumping Statements.

Module/ Chapter 3: Arrays and Strings
Arrays, Strings: C-Style Strings, String Class (<string>).

Module/ Chapter 4: Structures, Pointers, and Functions

Structures, Pointers, Functions: Basic Functions, Recursive Functions, Overloaded Functions, Inline Functions, Functions with Default Arguments.

Unit-II: Classes, Objects, and Operator Overloading

Module/ Chapter 5: Classes - I
Defining Classes and Creating Objects, Access Specifiers: Public, Private, and Protected Access, Type of Classes.

Module/ Chapter 6: Classes - II
Array and String as Data Members, Pointer to Objects, Constructors and its types, Overloaded Constructors, Destructors: Automatic Object Cleanup.

Module/ Chapter 7: Function Overloading, Operator Overloading, and Friend Functions
Function Overloading: Concept and Examples. Operator Overloading: Overloading Unary and Binary Operators. Friend Functions: Usage and Implementation.

Unit-III: Inheritance and Polymorphism

Module/ Chapter 8: Inheritance
Concept of Derived and Base Class, Accessing Base Class Members, Single Inheritance, Multiple Inheritance, Hierarchical Inheritance, Multilevel Inheritance, Hybrid Inheritance. Constructor in Derived Classes.

Module/ Chapter 9: Virtual Functions and Abstract Classes
Virtual Functions, Pure Virtual Functions, Abstract Classes.

Module/ Chapter 10: Polymorphism
Polymorphism: Compile-time and Runtime Polymorphism, Virtual Base Classes.

Unit-IV: Exception Handling, File Handling and Standard Template Library

Module/ Chapter 11: Exception Handling
Basics of Exceptions, Handling Exceptions using try, catch, and throw, Multiple Catch Blocks and Catch-All Handlers.

Module/ Chapter 12: File Handling
Classes for File Stream Operations (ifstream, ofstream, fstream), File Operations (Opening and Closing Files, File Modes), Detecting End-of-File, Sequential Input and Output Operations. File Pointers (Manipulation using seekg, seekp, tellg, tellp), Updating Files and Error Handling.

Module/ Chapter 13: Standard Template Library (STL) -I
Overview of STL, definition and importance, advantages of using STL in C++, components of STL – containers, iterators, and algorithms. Containers in STL, concept of generic containers.

Module/ Chapter 14: Standard Template Library (STL) -II

Types of containers – sequence containers (vector, list, deque), associative containers (set, map), and derived containers (stack, queue). Basic operations with STL.

Suggested Readings:

- E. Balaguruswamy, Object-Oriented Programming with C++, 8th Edition, McGraw Hill, 2020.
- Herbert Schildt, C++: The Complete Reference, 4th Edition, McGraw Hill, 2002.
- R.S. Salaria, Object-Oriented Programming with C++, 6th Edition, Khanna Book Publishing, 2021.
- Robert Lafore, Object-Oriented Programming in C++, 4th Edition, Sams Publishing, 2001.
- Bjarne Stroustrup, The C++ Programming Language, 4th Edition, Addison-Wesley, 2013.
- Stanley B. Lippman, Josee Lajoie, Barbara E. Moo, C++ Primer, 5th Edition, Addison-Wesley, 2012.
- Scott Meyers, Effective C++: 55 Specific Ways to Improve Your Programs and Designs, 3rd Edition, Addison-Wesley, 2005.
- Stephen Prata, C++ Primer Plus, 6th Edition, Addison-Wesley, 2011.
- Nicolai M. Josuttis, The C++ Standard Library: A Tutorial and Reference, 2nd Edition, Addison-Wesley, 2012.
- Bruce Eckel, Thinking in C++, 2nd Edition, Prentice Hall, 2000.
- Andrew Koenig, Barbara Moo, Accelerated C++: Practical Programming by Example, 1st Edition, Addison-Wesley, 2000.

Course Code: **MCA 403**
Name of Course: **Fundamentals of Computer & Programming in C**
Course Coordinator: **Mr. Mayank Chopra**
Credit: **4**
Course Objectives:

- To introduce students to the fundamental concepts of computers and their applications.
- To familiarize students with the architecture and working of computer systems.

- To develop an understanding of number systems, memory organization, and storage devices.
- To provide insights into input/output devices, software concepts, and networking fundamentals.
- To equip students with foundational programming skills in C language and build problem-solving abilities.

Course Outcomes: After the successful completion of this course, the student will be able to

CO[1]- Demonstrate a clear understanding of computer fundamentals, including their evolution, characteristics, and limitations.

CO[2]- Explain the architecture of computer systems and their components.

CO[3]- Perform number system conversions and apply memory organization concepts in computing solutions.

CO[4]- Write, compile, and debug C programs using control structures, functions, and arrays to solve computational problems.

Evaluation Criteria:
A student will be evaluated on the following basis:
End Semester Examination: 70% weightage
Continuous Internal Assessment: 30% weightage

Unit-I: Computer Fundamentals and Number Systems

Module/ Chapter 1: Introduction to Computers

Definition and Characteristics of Computers, Evolution and Generations of Computers, Capabilities and Limitations of Computer Systems

Module/ Chapter 2: Computer System Architecture

Block Diagram of a Computer, Components: Input Unit, Output Unit, Storage Unit, ALU, Control Unit, CPU

Module/ Chapter 3: Number Systems and Conversions

Non-Positional vs. Positional Number Systems, Decimal, Binary, Octal, and Hexadecimal Systems, Conversion Techniques between Number Systems

Module/ Chapter 4: Memory Organization and Storage Devices

Main Memory: RAM, ROM, PROM, EPROM, Cache, Memory Capacity and Organization, Secondary Storage Devices: Magnetic Tape, Disks, Optical Disks (CD-ROM, CDR, CD-RW, DVD)

Unit-II: Computer Peripherals, Software, and Networking

Module/ Chapter 5: Input Devices

Keyboard, Pointing Devices (Mouse, Touch Screen, Joystick, Trackball, Electronic Pen), Scanning Devices: Optical Scanners, OCR, OMR, Barcode Reader, MICR, Digital Cameras, Card Readers

Module/ Chapter 6: Output Devices

Monitors: CRT, LCD, Printers: Dot Matrix, Inkjet, Laser, Plotters, Screen Image Projectors

Module/ Chapter 7: Software Concepts

System Software vs. Application Software, Operating Systems and Utility Programs, Programming Languages: Machine, Assembly, High-Level, Translators: Assembler, Compiler, Interpreter

Module/ Chapter 8: Data Communication and Networks

Elements of Communication Systems, Transmission Modes: Simplex, Half-Duplex, Full Duplex, Data Transmission Speeds: Narrowband, Voice Band, Broadband, Transmission Media: Twisted Pair, Coaxial Cable, Optical Fibers, Network Types: LAN, WAN, MAN, Internet, WWW, Web Browsers, Modems

Unit-III: Basics of C Programming

Module/ Chapter 9: Introduction to C Programming

General Structure of a C Program, Compilation, Editing, and Running of C Programs, Data Types, Constants, Variables, Operators, and Expressions, Storage Classes and Type Casting, Unformatted and Formatted I/O Functions

Module/ Chapter 10: Control Structures in C

Decision Making: if, if-else, elseif, switch Statements, Looping Constructs: for, while, do-while, Program Control: break, continue, return Statements

Module/ Chapter 11: Functions in C

Defining and Declaring Functions, Function Invocation and Argument Passing, Recursion, External, Static, and Register Variables, Block Structure and Initialization

Unit-IV: Advanced C Programming Concepts

Module/ Chapter 12: Arrays and Strings

Introduction to Arrays, Declaring, Initializing, and Processing Arrays, Pointers to Arrays, Passing Arrays as Arguments to Functions, Introduction to Strings and Pointers to Strings, Passing Strings and Arrays of Strings to Functions

Module/ Chapter 13: Pointers in C

Definition, Declaration, and Usage of Pointers, Pointer Arithmetic and Types of Pointers, Accessing Values via Pointers, Programming Examples Using Pointers

Module/ Chapter 14: Structures and File Handling

Declaring and Initializing Structures, Accessing Elements of Structures, Arrays of Structures, Nested Structures, Pointers to Structures and their Usage, File Handling Concepts: Opening, Closing, Reading, and Writing Files, Working with Structures in Files

Suggested Readings:

- Pradeep K. Sinha, PritiSinha, “Computer Fundamentals”, 6E , BPB Publications
- Rajaraman, V., “Fundamental of Computers”, Fifth Edition, Prentice Hall India, New Delhi.

- E. Balagurusamy, “Introduction to Computers (Special Indian Edition)”, Tata McGraw Hill.
- R S Salaria, Application in C, Khanna book publishing
- E. Balagurusamy, “Programming in ANSI C”, 8E ,Tata McGraw Hill
- YashwantKanetakar, “Let us C” BPB.

Course Name: **Indian Knowledge System**

Course Code: **IKS**

Course Coordinator: Prof. Bhag Chand Chauhan

Credits: **02**

Course Objectives: To introduce the students to the Bhartiya knowledge system which encapsulates the ancient/traditional education system and various schools of

philosophy and to acquaint them with various technological and scientific endeavours in the Bhartiya *parampara*.

Course Outcomes:

- The course will delve into the antiquity of the Bhartiya civilization.
- It will make learners help build a deeper appreciation of knowledge attainments of India since earliest times.
- How the Bhartiya Knowledge System encompassed varied dimensions of human life will be learned.
- The course will develop an interdisciplinary acumen in the learners as it will deal with Bhartiya sciences, art and architecture.
- The course will develop the feeling of pride vis a vis the nation as the knowledge provider of the world.

Evaluation Criteria:

1. Continuous Internal Assessment (CIA): 30%
2. External End Semester Examination (ESE): 70%

Unit-I: **Development of Indian Knowledge Systems**

Module/Chapter 1: **History and Foundation of IKS**

Definition, Concept, and Scope of IKS, Genesis of the land, Discovery of the Saraswatī River, Rakhigarhi Site

Module/Chapter 2: **Learning Centers and Knowledge Proliferation**

Ancient Education Institutes e.g. Takṣaśilā, Nālandā, Vikramśilā, and Vallabhi, Alumni, Trade Routes and Networks for Knowledge Export from Bhārata.

Unit-II: **Creation of Art and Literature**

Module/Chapter 3: **Art Forms and Literature**

Forms of Arts, Naṭarāja– A Masterpiece of Bhāratīya Art, the Vedas, Up-Vedas, Vedangas, Schools of Philosophy, Vedanta

Module/Chapter 4: **Scholars and Their Works**

Works of Vālmīki, Yājňavalkya-Gārgī Debate, Bodhāyana, Kaṇāda, Patañjali, Āryabhaṭa, Bramhgupta, Bhāskarācārya, and Neelkantha.

Unit-III: **Matter and Astronomy**

Module/Chapter 5: **Nature of Matter**

Kanada Theory of Matter and Concept Gravity, Sage Agastya's Model of Battery, Types of Motion

Module/Chapter 6: **Mathematics and Astronomy**

Decimal System, Vedic Mathematics, Kerala School for Mathematics and Astronomy, History and Culture of Astronomy, Archaeostronomy, and Calendars.

Unit-IV: **Materials and Technology**

Module/Chapter 7: **Materials and Products**

Juices, Dyes, and Paints, Glass and Pottery, references form the Vedic Age and Post-Vedic Records

Module/Chapter 8: **Stone and Metal Marvels**

Classical Stone Marvels, Metallurgy, Marine Technology, Water Reservoirs.

Unit-V: **Plants, Health and Natural Healing**

Module/Chapter 9: **Plants and Environment**

Life Science in Plants, Agriculture-Archaeological Evidences and Vedic Period, Practices of Ecology and Environment

Module/Chapter 10: **Health and Healing**

Āyurveda, Types of Surgery, and Ashtanga Yoga

Suggested Readings

- D.M. Bose, S.N. Sen and B.V. Subbarayappa (Eds.) 1971. *A Concise History of Science in India*. Indian National Science Academy: New Delhi.
- Debiprasad Chattopadhyaya (Ed.) 1982. *Studies in the History of Science in India (2 Vols.)*. Editorial Enterprises: New Delhi.
- D. Frawley. 1994. "Planets in the Vedic literature", *Indian Journal of History of Science*. 29, 495-506.
- D.P. Agrawal. 2000. *Ancient Metal Technology and Archaeology of South Asia (A Pan-Asian Perspective)*, Aryan Books International: New Delhi.
- H.C. Bhardwaj. 1979. *Aspects of Ancient Indian Technology*, Munshiram Manoharlal: New Delhi.
- Kumar Biswas. 1996. *Minerals and Metals in Ancient India*, D.K. Printworld: New Delhi.
- R. Shrivastava. 2006. *Mining and Metallurgy in Ancient India*, Munshiram Manoharlal: New Delhi.

- O.P. Jaggi. 1977. *Science and Technology in Medieval India*. Atma Ram & Sons: Delhi.
- A.Y. Al-Hasan. 2001. *Science & Technology in Islam*. UNESCO.
- Ranabir Chakravarti. 2008. “Agricultural Technology in Medieval India”, *The Medieval History Journal*, Vol. 11, No. 2, July-December.
- D. Kumar. “Patterns of colonial science in India”. *Indian Journal of History of Science*, 15 (1), 105-113.
- Deepak Kumar. 1995. *Science & the Raj*, OUP: Delhi.
- S.B. Dikshit. 1931. *Bharatiya Jyotish Shastra*, Poona.
- A.F.R. Hoernle. 1907. *Studies in Medicine of Ancient India*, Oxford.
- Ian Inkstem. 1991. *Science & Technology in History*, London.
- V.V. Krishna and S.S. Bhatnagar. 1993. *Science, Technology and Development*, Delhi.
- A.K. Bisvas and K.L. Mukhopadhyaya. 1969. *Science in India*, Kolkata.
- Dhruv Raina. 2003. *Image and Context: Historiography of Science in India*, OUP: Delhi.
- B. C. Chauhan. 2023. IKS: The Knowledge System of Bharat, Garuda Prakashan, New Delhi

Course Name: Business Communication Skills
Course Code: MGT 528
Course Coordinator: Dr. Aditi Sharma
Credits: 02

Course Objectives: The course is designed to give learners the practical skills of business communication, stressing communication and non-communication communication along with overcoming workplace communication barriers. It stresses on significance of grooming, presentation and interview skills, body language and

professional email etiquettes, while promoting effective interpersonal skills through tools such as the Johari Window and transactional analysis.

Course Outcomes:

- The course will assist the learners in the acquisition of verbal and non-verbal communication skills in public speaking, group discussion, and interviewing.
- The course will enhance learners' understanding and use of non-verbal communication like body language, gestures, and postures, which will increase the capacity to communicate more than words in the workplace.
- The course will help learners appreciate the importance of personal grooming and business etiquette, including the proper use of mobile devices, emails, and telephone manners in the workplace.
- The course will enable the learners to comprehend the utilization of social networking and other internet-based platforms for business communication in personal and organizational communication strategies.
- learners will be able to recognize and use transactional analysis to enhance communication effectiveness, especially in business and interpersonal contexts.

Evaluation Criteria:

1. Continuous Internal Assessment (CIA): 30%
2. External End Semester Examination (ESE): 70%

Unit-I: Introduction to Business Communication

Module/ Chapter 1: Communication and its Importance in Business- Introduction, Meaning and Definition of Communication, Nature of Communication, Objectives of Communication, Importance of Communication, Factors affecting Communication, Communication Process, Barriers in the Communication Process, Strategies to Overcome Communication Barriers, Types of Communication, Importance of Verbal and Non-Verbal Communication in Business, Differences between Verbal and Non-Verbal Communication

Module/ Chapter 2: Use of Social Networks for Business Communication- Introduction, The Social Communication Model, The Emergence of Mobile as a Platform for Business Communication, Role of Social Media in Business Communication, Popular Social Media Platforms for Business, Advantages and Challenges of Using Social Networks, Best Practices for Effective Communication via Social Media

Unit-II: Verbal Communication Skills

Module/Chapter 3: Public Speaking- Introduction, Nature of Public Speaking, Importance of Public Speaking in Business, Key Principles of Effective Public Speaking, Common Mistakes in Public Speaking, Do's and Don'ts of Public Speaking, Techniques to Improve Public Speaking Skills

Module/Chapter 4: Presentation skills- Introduction, Essentials of an Effective Presentation, Structure of a Business Presentation, Designing Presentation Slides, Common Presentation Mistakes, Tips for Delivering an Engaging Presentation

Unit-III: Non-verbal Communication Skills

Module/Chapter 5: Body language, Gestures and Postures- Introduction, Meaning and Importance of Non-Verbal Communication, Types of Non-Verbal Communication, Importance of Non-Verbal Communication, Body Language and Its Role in Communication, Gestures and Postures in Business Communication, Proxemics

Module/Chapter 6: Grooming and Grooming Tips- Introduction, Importance of Personal Grooming in Professional Life, Elements of Professional Grooming. Grooming Tips for Workplace Success, Etiquettes: Meaning and Importance of Business Etiquette, Types of Business Etiquette, Workplace Etiquette and Social Etiquette, Email and Telephone Etiquette, Etiquette for Mobile Device Use, Business Etiquette in Social Context

Unit-IV: Written Communication Skills

Module/Chapter 7: Writing Effective E-mails- Introduction, Importance of Email Communication, Structure of a Professional Email, Common Mistakes in Email Writing, Best Practices for Effective Email Writing, Managing Email Overload and Enhancing Inbox Efficiency, Ensuring Confidentiality and Privacy in Email Exchanges, Approaches for Effective Communication in a Remote Work Setting

Module/Chapter 8: Resume Writing- Introduction, Importance of a Well-Written Resume, Types of Resumes - Chronological, Functional, and Combination, Essential Components of a Resume- Structure, Length, and Content, Format and Design, Integrity, The Role of Keywords in ATS (Applicant Tracking Systems), Common Resume Mistakes and How to Avoid Them

Unit-V: Interpersonal and Interview skills

Module/Chapter 9: Transactional Analysis and Effective Communication- Introduction, Meaning and Importance of Transactional Analysis, Components of Transactional Analysis, Role of Transactional Analysis in Effective Communication, Application of Transactional Analysis in the Workplace

Module/Chapter 10: Interviews skills- Introduction, Importance of Interview Skills in Career Growth, Types of Interviews - Personal, Telephonic, Virtual, Common Interview Questions and How to Answer Them, Do's and Don'ts in an Interview, Post-Interview Etiquette

Suggested Readings:

1. Newman, A. (2022). Business Communication and Character. United States: Cengage South-Western.
2. Bovée, C. L., & Thill, J. V. (2020). Business communication essentials., 8th Edition, Pearson Education Ltd., UK
3. Kaul, A. (2009). *Business Communication*. PHI Learning Pvt. Ltd..
4. Raman, Meenakshi. Prakash Singh (2013), *Business Communication*, Oxford University Press, New Delhi, Second Edition.
5. Mukerjee, Hory Sankar (2013). *Business Communication*, Oxford University Press, New Delhi.
6. Sengupta, Sailesh. (2011) Business and Managerial Communication. PHI Learning Pvt. Ltd.
7. Koneru, Aruna. (2008) Professional Communication. Mcgraw Hill Education. 1st Edition.
8. Davis Kenneth W. (2008) Business Writing and Communication. The Mcgraw Hill Education. 1st Edition.

9. Guffey, Mary Ellen. Richard Almonte. (2013). Essentials of Business Communication, Nelson, Sixth Canadian.
10. Raina, Roshan Lal. Iftikhar Alam. Faizia Siddiqui. (2010). Communication for Management. Word Press, New Delhi
11. Indrajit Bhattacharya, An Approach to Communication Skills, Delhi: Dhanpat Rai, 2008.

SEMESTER II

Course Code: **MCA 504**
Name of Course: **Theory of Computation**
Course Coordinator: **Mr. Mayank Chopra**
Credit: **4**

Course Objectives:

- To introduce students to the fundamental concepts of automata theory and computation.

- To familiarize students with finite automata, formal languages, and their applications.
- To develop an understanding of context-free grammars, pushdown automata, and Turing machines.
- To provide insights into computational complexity and problem classification.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1- Demonstrate an understanding of the fundamental concepts of automata theory, including alphabets, strings, and finite automata.

CO2- Analyze and design deterministic and non-deterministic finite state machines, Mealy and Moore machines, and apply automata minimization techniques.

CO^3- Apply formal language concepts to classify languages using Chomsky's hierarchy, derive context-free grammars, and identify ambiguous grammars.

CO^4- Describe the functioning of pushdown automata and Turing machines and apply computational complexity concepts to classify problems into P, NP, NP-Complete, and NP-Hard categories.

Evaluation Criteria:
A student will be evaluated on the following basis:
End Semester Examination: 70% weightage
Continuous Internal Assessment: 30% weightage

Unit-I: Finite Automata and Machines

Module/ Chapter 1: Introduction to Theory of Computation
Overview of Theory of Computation, Definitions: Alphabet, Strings, and their properties, Introduction to Automata Theory

Module/ Chapter 2: Finite Automata Basics
Definition of Finite Automata, Description of Finite Automata, Transition Graphs and Transition Functions, Acceptability of Strings by Finite Automata

Module/ Chapter 3: Deterministic & Non-Deterministic FSMs
Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NDFA), Equivalence of DFA and NDFA, Conversions between DFA and NDFA

Module/ Chapter 4: Mealy & Moore Machines & Minimization
Introduction to Mealy and Moore Machines, Designing of Mealy and Moore Machines, Minimization of Finite Automata, Applications of Automata Minimization

Unit-II: Formal Languages and Grammars

Module/ Chapter 5: Chomsky Classification of Languages
Language Hierarchies, Types of Grammars: Type 0, Type 1, Type 2, Type 3, Relationship between Languages and Automata

Module/ Chapter 6: Regular Languages & Expressions

Definition of Regular Sets, Regular Expressions and Their Properties, Finite State Machines and Regular Expressions, Applications of Regular Expressions

Module/ Chapter 7: Pumping Lemma & Closure Properties

Pumping Lemma for Regular Sets, Applications of Pumping Lemma, Closure Properties of Regular Sets

Module/ Chapter 8: Context-Free Grammars & Languages

Introduction to Context-Free Grammars (CFG), Context-Free Languages (CFLs) and Derivation Trees, Ambiguity in CFGs, Simplification of CFGs, Normal Forms: Chomsky Normal Form & Greibach Normal Form

Unit-III: Pushdown Automata (PDA)

Module/ Chapter 9: Introduction to Pushdown Automata (PDA)

Definition and Representation of PDA, Moves and Instantaneous Descriptions, Acceptance by PDA: Empty Stack & Final State

Module/ Chapter 10: Deterministic Pushdown Automata (DPDA)

Definition and Characteristics of DPDA, Difference between PDA and DPDA, Limitations and Capabilities of DPDA

Module/ Chapter 11: PDA & CFG Equivalence

Construction of PDA from CFG, Construction of CFG from PDA, Equivalence of PDAs and CFGs

Module/ Chapter 12: Closure Properties of CFLs

Closure Properties of Context-Free Languages, Decision Properties of CFLs, Comparison with Regular Languages

Unit-IV: Turing Machines & Computational Complexity

Module/ Chapter 13: Turing Machines & Recursive Languages

Introduction to Turing Machines (TM), Design and Representation of TMs, Universal Turing Machines (UTM), Church's hypothesis, Recursive and Recursively Enumerable Languages

Module/ Chapter 14: Complexity Theory

Tractable and Intractable Problems, P, NP, NP-Complete, and NP-Hard Problems

Suggested Readings:

- John E. Hopcroft, Jeffery Ullman, "Introduction to Automata theory, Languages & computation" , Narosa Publishers
- John C Martin, "Introdution to languages and theory of computation", McGraw Hill
- K.L.P Mishra & N.Chandrasekaran, "Theory of Computer Science", PHI Learning
- Daniel I.A. Cohen, "Introduction to Computer Theory", Wiley India.
- Peter Linz, "An Introduction to Formal Languages and Automata", Jones & Bartlett Learning

Course Code : MCA 505

Course Title : Database Management System

Course Coordinator : Mr. Manoj Dhiman

Credit : 4

Course Objectives:

- To introduce core database concepts, including data models, file organization techniques, and database architectures, to build a strong foundation in database management.
- To develop the ability to design relational databases using Entity-Relationship (ER) modelling and apply normalization techniques to ensure data consistency and efficiency.
- To provide hands-on experience in writing and optimizing SQL queries for data definition, manipulation, and retrieval while maintaining integrity constraints.
- To cover essential transaction management concepts, concurrency control mechanisms, and database recovery techniques to ensure data reliability and security in multi-user environments.

Course Outcomes:

After successful completion of this course, students will be able to:

- **CO^1** – Describe fundamental database concepts, different data models, and database architectures, and evaluate file organization techniques for efficient data storage.
- **CO^2** – Design well-structured Entity-Relationship (ER) models and translate them into normalized relational schemas suitable for implementation.
- **CO^3** – Write and execute SQL queries to create, retrieve, update, and delete data while ensuring data integrity and security.
- **CO^4** – Apply normalization techniques to optimize database performance by reducing redundancy and improving consistency.
- **CO^5** – Implement transaction management techniques, concurrency control mechanisms, and database recovery strategies to ensure data reliability and consistency.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Unit-I: Database Concepts and File Organization

Module/ Chapter 1: Basic Concepts of Databases

Database, DBMS, Entity, Attributes, Domain, Relationship, Components and Structure of a Database, Three-level Architecture of DBMS, Advantages and Disadvantages.

Module/ Chapter 2: Data Models
Database models (Hierarchical, Network, Relational, ER, Object-Oriented, Document).

Module/ Chapter 3: File Organization
Serial File Organization, Sequential File Organization, Index File Organization, Direct File Organization.

Unit-II: Entity-Relationship Model, Relational Data Model, and Relational Database Design

Module/ Chapter 4: Entity-Relationship Model
Entity Types, Entity Sets, Attributes & Keys, Relationships, Relationship Types, Design Issues in E-R models, E-R Diagrams, Design of an E-R Database Schema, Reduction of an E-R Schema to Tables.

Module/ Chapter 5: Relational Data Model - I
Relational Model Concepts, Integrity Constraints over Relations, Relational Algebra.

Module/ Chapter 6: Relational Data Model – II
Relational Calculus (Tuple Relational Calculus, Domain Relational Calculus).

Module/ Chapter 7: Relational Database Design
Functional Dependencies, Decomposition, Normal Forms based on Primary Keys (1NF, 2NF, 3NF, BCNF), Multi-valued Dependencies, Fourth Normal Form (4NF), Join Dependencies, Fifth Normal Form (5NF).

Unit-III: SQL

Module/ Chapter 8: Data Types and Operators
Data Types, Operators.

Module/ Chapter 9: SQL Languages and Constarints
SQL Languages, Constraints

Module/ Chapter 10: Queries, Indexing and Views
Queries in SQL, Indexes in SQL, Views.

Module/ Chapter 11: Functions
Built-in Functions (Date/Time, Numeric, Aggregate, and String Functions).

Unit-IV: Transaction Processing, Concurrency Control and Recovery

Module/ Chapter 12: Transaction Processing Concepts
Introduction to Transaction Processing, Transaction & System Concepts, Schedules.

Module/ Chapter 13: Concurrency Control Techniques
Introduction to Concurrency Control, Locking Techniques,Timestamp Ordering, Multi-version Techniques, Optimistic Techniques, and Granularity Level Locking Techniques.

Module/ Chapter 14: Recovery Techniques

Introduction to Recovery, Recovery Concepts, Recovery Techniques in Centralized DBMS

Suggested Readings:

- Elmasri & Navathe, Fundamentals of Database Systems, 7th Edition, Pearson, 2016.
- Ramez Elmasri, Shamkant B. Navathe, Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition, Pearson, 2016.
- Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 7th Edition, McGraw Hill, 2019.
- C. J. Date, An Introduction to Database Systems, 8th Edition, Pearson, 2003.
- Raghu Ramakrishnan, Johannes Gehrke, Database Management Systems, 3rd Edition, McGraw Hill, 2003.
- Peter Rob, Carlos Coronel, Database Systems: Design, Implementation, and Management, 12th Edition, Cengage Learning, 2018.
- R. S. Salaria, Database Management Systems, 2nd Edition, Khanna Publishing, 2021.
- T. Connolly, C. Begg, Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition, Pearson, 2014.

Course Code : MCA 506

Course Title : Database Management System Lab

Course Coordinator : Mr. Manoj Dhiman

Credit : 2

Course Objectives:

- To provide hands-on experience with SQL and database management operations.
- To implement database creation, table manipulation, and querying techniques using SQL.
- To explore advanced SQL concepts, including joins, subqueries, indexes, and views.
- To develop proficiency in writing stored procedures, triggers, and transactions.

Course Outcomes: Upon successful completion of this course, students will be able to:

- **CO^1** – Create and manage relational databases using SQL commands to store and organize structured data efficiently.
- **CO^2** – Perform CRUD (Create, Read, Update, Delete) operations using SQL queries while ensuring data integrity and security.
- **CO^3** – Implement constraints, joins, and functions to extract meaningful insights from relational data.
- **CO^4** – Write and optimize complex queries using subqueries, views, and indexing techniques to improve query performance.
- **CO^5** – Design and implement stored procedures, triggers, and transactions to automate and maintain database operations in real-world scenarios.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Unit-I: Introduction to SQL

Module/ Chapter 1: Basics of SQL
Overview of SQL, SQL syntax, Data Types (numeric types: INT, FLOAT, DECIMAL; character types: CHAR, VARCHAR, TEXT; date/time types: DATE, TIME, TIMESTAMP; and other types like BOOLEAN, BLOB), SQL commands (DDL, DML, DCL, TCL), creating and managing databases.

Module/ Chapter 2: Working with Tables
Creating tables, inserting, updating, and deleting data, modifying table structures using ALTER, and dropping tables.

Module/ Chapter 3: Basic Query Operations
SELECT, WHERE, ORDER BY, DISTINCT, LIMIT, filtering and sorting data.

Unit-II: Operators and Constraints

Module/ Chapter 4: Operators
Arithmetic Operators, Comparison Operators, Logical Operators, Bitwise Operators, Assignment Operator, String Operators, Set Operators, Like Operators, Null Operators.

Module/ Chapter 5: Integrity Constraints
Column level and Table level constraints (primary key, foreign key, unique, not null, check, default constraints).

Unit-III: Functions, Joins and Subqueries

Module/ Chapter 6: Functions - I
String Functions, Numeric Functions, Date and Time Functions, Control Flow Functions.

Module/ Chapter 7: Functions - II
Aggregate Functions, Bitwise Functions, Encryption Functions, Type Conversion Functions.

Module/ Chapter 8: Joins and Subqueries
Inner Join, Outer Join, Self-Join, Cross Join, Correlated And Non-Correlated Subqueries.

Unit-IV: Advanced SQL Concepts

Module/ Chapter 9: Indexes, Views and Transactions
Creating and using indexes, Creating and managing views, working with transactions (COMMIT, ROLLBACK, SAVEPOINT).

Module/ Chapter 10: Stored Procedures, Functions, and Triggers
Creating and executing stored procedures, user-defined functions, working with triggers (BEFORE, AFTER, INSTEAD OF).

Suggested Readings:

- James R. Groff, Paul N. Weinberg, Andy Oppel, SQL: The Complete Reference, 3rd Edition, McGraw Hill, 2011.
- Robert Sheldon, Geoff Moes – Beginning SQL, 1st Edition, Wrox, 2005.
- Martin Gruber, Mastering SQL, 1st Edition, O'Reilly Media, 1990.
- Paul DuBois, MySQL Cookbook: Solutions for Database Developers and Administrators, 3rd Edition, O'Reilly Media, 2014.
- Alan Beaulieu, Learning SQL: Master SQL Fundamentals, 3rd Edition, O'Reilly Media, 2020.
- Joel Murach, Murach's MySQL, 2nd Edition, Mike Murach & Associates, 2019.

- Kevin Loney, Oracle Database 12c SQL, 1st Edition, McGraw Hill, 2014.
- Joe Celko – Joe Celko's SQL for Smarties: Advanced SQL Programming, 5th Edition, Morgan Kaufmann, 2015.

Course Code: MCA 507

Name of Course: Design and Analysis of Algorithms

Course Coordinator: Mr. Ajay Kumar

Credit: 2

Course Objectives:

- To equip students with the ability to understand, design, and implement efficient algorithms for a wide range of computational problems**.**
- To teach students various algorithm design paradigms, including divide-and-conquer, greedy algorithms, dynamic programming, enabling them to choose the most appropriate approach for a given problem.
- To train students in the analysis of algorithms, including determining time and space complexity using asymptotic notations (Big O, Omega, Theta), and evaluating algorithm performance in different scenarios.
- To empower students to apply algorithmic principles to real-world software development challenges, leading to the creation of optimized and efficient applications.

Course Outcomes: After the successful completion of this course, the student will be able to

- Learn different algorithm design techniques such as Divide and Conquer, Greedy Approach, and Dynamic Programming to solve complex computational problems efficiently.
- Develop skills to evaluate the efficiency of algorithms by analyzing their time and space complexities using Big O, Omega, and Theta notations.
- Gain expertise in optimizing algorithms to improve performance, reduce execution time, and ensure scalability in real-world applications.
- Apply various algorithmic strategies in domains like graph theory, searching, sorting, and numerical computations to solve industry-related challenges.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content:

Unit-I: Introduction

Module/ Chapter 1: **Analysis of Algorithms**

Definition and Characteristics of algorithms, examples on asymptotic notations, and time complexity

Module/ Chapter 2: **Design of Algorithms**

Design Techniques, Comparison based searching and sorting algorithms, and linear time sorting algorithms

Module/ Chapter 3: **Heap and heap sort**

Concept of heap and its operations, binary heap, heap sort, and running time analysis

Unit-II: Recurrences and Divide and Conquer

Module/ Chapter 4: **Solving Recurrences**

Problems on Substitution method, and Master Theorem

Module/ Chapter 5: **Divide and Conquer**

General method, Binary search, Merge Sort, Quick sort analysing of time and Space Complexity

Unit-III: Graph and Greedy Strategy

Module/ Chapter 6: **Graph Theory**

Concept of graphs, Directed and undirected graphs, Breadth First Search, and Depth First Search

Module/ Chapter 7: **Greedy techniques I**

Introduction to the Greedy algorithm, examples of greedy method like fractional knapsack problem, Dijkstra's Algorithm

Module/ Chapter 8: **Greedy techniques II**

Concept of Spanning tree and its properties, Minimum spanning tree and its properties, algorithm to find minimum spanning tree: Prim’s algorithm, Kruskal algorithm

Unit-IV: Dynamic Programming

Module/ Chapter 9: **Dynamic Programming I**

Concept of dynamic programming, problems based on this approach such as Fibonacci numbers, and 0/1 knapsack problems

Module/ Chapter 10: **Dynamic Programming II**

Matrix Chain Multiplication, Floyd-Warshall algorithm

Suggested Readings:

1. Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., & Stein, Clifford. Introduction to Algorithms. MIT Press. 3rd ed. 2009.

2. Sarabasse& A.V. Gelder Computer Algorithm – Introduction to Design and Analysis, Publisher – Pearson 3rd Edition 1999
3. Sedgewick, Robert, & Wayne, Kevin. Algorithms. Addison-Wesley Professional. 4th ed. 2011.
4. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
5. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.

Course Code: **MCA 522**

Name of Course: **Software Testing**

Course Coordinator: **Dr. Pankaj Kumar**

Credit: **02**

Course Objectives:

1. To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
2. Analysis of various testing methodologies and procedure to design test cases.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1- Analyze requirements to determine appropriate testing strategies.

CO^2 - Apply a wide variety of testing techniques in an effective and efficient manner.

CO^3 - Compute test coverage and yield according to a variety of criteria.

CO^4 - Evaluate the limitations of a given testing process and provide a succinct summary of those limitations.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70%weightage
- Continuous Internal Assessment: 30% weightage

Course Content

UNIT - I: **Software Testing**

Module/ Chapter 1: **Fundamentals of Software Testing**

Definition and Importance of Software Testing, Goals and Objectives of Testing, Software Testing Principles, Verification and validation, Test plan and test case specification

Module/ Chapter 2 **Levels of Testing**

Unit testing, Integration testing (Top-down and Bottom-up approaches), Alpha and Beta testing, System testing, Acceptance testing

Module/ Chapter 3 **Testing Strategies and Debugging**

System testing and debugging, Functional testing vs. Structural testing, Software testing strategies

Unit-II: Software Maintenance

Module/ Chapter 4 Fundamentals of Software Maintenance

Definition and Importance, Challenges in Maintenance, Types of Software Maintenance, Structured vs. Unstructured maintenance, Maintenance models

Module/ Chapter 5 Configuration and Change Management

Configuration management, Reverse engineering, Software Re-engineering, Importance and benefits of software re-engineering, Re-engineering Process Models

Unit-III: Software Project Management

Module/ Chapter 6 Project Planning and Scheduling

Fundamentals of project planning, Project scheduling techniques Gantt Charts, Critical Path Method (CPM), Program Evaluation and Review Technique (PERT)

Module/ Chapter 7 Software Metrics and Cost Estimation

Size metrics (LOC, Token Count, Function Count), Cost estimation models (COCOMO)

Module/ Chapter 8 Risk Management in Software Projects

Risk management activities, Identifying and mitigating risks in software projects

Unit-IV: Software Reliability and Quality Assurance

Module/ Chapter 9 Software Reliability and Software Quality Assurance (SQA)

Reliability issues in software development, Reliability metrics and models

Module/ Chapter 10 Software Quality Assurance (SQA)

Importance of software quality, Quality assurance practices in software development

Suggested Readings

I. Aditya Mathur, Foundations of Software Testing, Pearson India

II. Roger S Pressman Software Engineering: A Practitioner's Approach, 9 edition, McGraw-Hill Higher Education

Suggested Books:

I. Paul C. Jorgensen, Software Testing: A Craftsman's Approach.
II. S. Desikan & G. Ramesh, Software Testing: Principles and Practices.

Course Code:	**MCA 527**
Name of Course:	**Python Programming**
Course Coordinator:	**Mr. Mayank Chopra**
Credit:	**2**

Course Objectives:

- To introduce students to the fundamental concepts of Python programming.
- To develop problem-solving abilities using Python control structures and data types.
- To familiarize students with functions, modules, and packages for modular programming.
- To equip students with the knowledge of file handling, exception management, and object-oriented programming in Python.

Course Outcomes: After the successful completion of this course, the student will be able to

CO[1]- Demonstrate an understanding of Python programming fundamentals, including its history, environment setup, syntax, and program execution.

CO2- Apply Python control structures and native data types to write efficient and logical programs **.**

CO[3]- Develop modular and reusable code using functions, modules, and packages in Python.

CO[4]- Implement file handling operations, manage exceptions, and apply object-oriented programming concepts for real-world problem-solving**.**

Evaluation Criteria:

A student will be evaluated on the following basis:

End Semester Examination: 70% weightage

Continuous Internal Assessment: 30% weightage

Unit-I: Introduction to Python Programming

Module/ Chapter 1: Introduction to Python

History and Origin of Python Language, Installing Python and Setting up Environment Variables, Running Python Programs, Writing and Executing Your First Python Program

Module/ Chapter 2: Python Basics

Keywords, Identifiers and Statements, Indentation and Documentation, Variables and Multiple Assignment

Module/ Chapter 3: Data Types and Input/Output

Python Data Types and Type Conversion, Input and Output Functions, Using the Import Command

Module/ Chapter 4: Operators and Expressions

Types of Operators in Python, Expressions and Evaluations, Operator Precedence and Associativity

Unit-II: Control Structures and Native Data Types

Module/ Chapter 5: Control Structures

Decision-Making Statements (if, if-else, nested if), Loops in Python (for and while), Control Statements (break, continue, pass)

Module/ Chapter 6: Python Native Data Types

Numbers, Lists, Tuples, Sets, and Dictionaries, Functions and Methods for Data Types, String Manipulation and Operations

Unit-III: Functions and Modules

Module/ Chapter 7: Functions in Python

Built-in and User-Defined Functions, Anonymous Functions (Lambda Functions), Recursion and Scope of Variables, Pass by Value vs. Pass by Reference

Module/ Chapter 8: Modules and Packages

Introduction to Python Modules and Packages, Creating and Importing Modules, Path Searching and Module Reloading, Standard Python Modules

Unit-IV: Exception Handling, File Management, and OOP

Module/ Chapter 9: Exception Handling

Understanding Exceptions and Their Types, Exception Handling using Try-Except Blocks, User-Defined Exception

Module/ Chapter 10: File Management and OOP Concepts

File Operations: Open, Read, Write, Close, File Methods: tell(), seek(), Renaming, Deleting Files, and Managing Directories, Object-Oriented Programming in Python: Classes and Objects, Class and Instance Attributes, Garbage Collection

Suggested Readings:

- R. S. Salaria, “Programming in Python”, Khanna Publishing
- A. Martelli, A. Ravenscroft, S. Holden, “Python in a Nutshell”, OREILLY
- R. NageswaraRao, “Core Python Programming”, 2E, Dreamtech.
- Martin C. Brown, “Python, The complete Reference”, McGraw Hill

Course Code: **MCA 551**
Name of Course: **IT Tool for IKS**
Course Coordinator: **Prof. Pradeep Chouksey**

Credit: **2**

Course Objectives:

- ✓ Learn the fundamentals of computer science and their integration with IKS.
- ✓ Apply Vedic mathematics and Jyotish principles in computing and data science.
- ✓ Explore AI, machine learning, and IT applications inspired by Yoga, Ayurveda, and Vastu.
- ✓ Utilize cloud computing and digital tools for preserving ancient Indian texts and knowledge.

Course Outcomes (COs)

Upon successful completion of the course, students will be able to:

1. **CO1**: Explain the significance and historical evolution of the Indian Knowledge System and its influence on various disciplines.
2. **CO2**: Apply principles of Vedic mathematics for faster computation, optimization techniques, and algorithmic problem-solving in IT.
3. **CO3**: Analyze the role of Indian astronomy (Jyotish) in data science, big data analysis, and AI-driven predictive modeling.
4. **CO5**: Utilize cloud computing and digital storage technologies to preserve and enhance the accessibility of ancient Indian scriptures and texts.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content

Unit-I: Introduction to Indian Knowledge System (IKS)

Module/ Chapter 1: Definition Historical Perspective of IKS
IKS refers to India's ancient wisdom in science, mathematics, medicine, architecture, and philosophy. Rooted in holistic knowledge with sustainable and ethical approaches, Applied in fields like healthcare (Ayurveda), architecture (Vastu), and computation (Vedic Math).

Module/ Chapter 2: Major Texts and Scriptures
Vedas – Oldest texts covering cosmology, rituals, and philosophy, Upanishads – Spiritual and philosophical knowledge on self-realization, Puranas – Historical

narratives, cosmology, and moral stories, Ayurveda – Holistic medical science focusing on natural healing, Vastu Shastra – Ancient architecture principles used in construction, Jyotish (Astrology) – Astronomical calculations for predictions and guidance Yoga – Mind-body discipline for physical and mental well-being.

Module/ Chapter 3: India's Contributions to Mathematics and Science
Aryabhata – Concept of zero, π (pi), and planetary motion, Brahmagupta – Rules for negative numbers, algebraic equations, Bhaskaracharya – Early calculus, advanced trigonometry, Srinivasa Ramanujan – Infinite series, number theory, modular functions.

Unit-II: Fundamentals of Computer Science

Module/ Chapter 4: Introduction to Computers and Their Evolution
From abacus to AI, computers evolved from manual calculations to automation, Key components – Hardware, software, networks, AI, and data processing.

Module/ Chapter 5: Vedic Mathematics and Computer Algorithms
Vedic math principles enhance fast calculations, cryptography, and optimization, Used in machine learning models, numerical computing, and algorithm efficiency.

Module/ Chapter 6: Indian Astrology and Pattern Recognition in Data Science
Jyotish principles involve big data analysis and probability, AI models use pattern recognition to analyze celestial data for forecasting.

Unit-III: Computer Science and IKS

Module/ Chapter 7: Yoga and Its Applications in Artificial Intelligence (AI)
AI-based wellness apps use biometric data to improve mental health tracking, Meditation and deep learning used to train AI for emotion recognition.

Module/ Chapter 8: Vastu Shastra and Computer-Based Simulations/Modeling
AI and simulation tools model optimal architectural designs based on Vastu, Used in urban planning, energy-efficient buildings, and real estate.

Module/ Chapter 9: Ayurveda, Digital Healthcare, and Machine Learning
AI-based dosha analysis for personalized medicine, Machine learning aids in diagnosis, herbal medicine recommendations, and predictive healthcare.

Module/ Chapter 10: Cloud Computing and Digital Storage of Ancient Indian Literature
Sanskrit texts, scriptures, and manuscripts are digitized using AI and blockchain, Cloud platforms enable global accessibility and preservation of Indian heritage.

Suggested Readings

1. **Kapil Kapoor & Avadhesh Kumar Singh** – *Indian Knowledge Systems* (D.K. Printworld, 2005)
2. **Bharati Krishna Tirthaji** –*Vedic Mathematics* (Motilal Banarsidass, 2010)

3. **Bibhutibhushan Datta & Avadhesh Narayan Singh** – *A History of Hindu Mathematics* (Asia Publishing House, 1962)
4. **Vasant Lad** – *Ayurveda: The Science of Self-Healing* (Lotus Press, 1984)
5. **B. B. Puri** – *Vastu Shastra: The Ancient Indian Science of Architecture and Design* (Diamond Books, 2005)
6. **Eknath Easwaran** – *The Essence of the Upanishads* (Nilgiri Press, 2009)
7. **David Frawley** – *Foundations of Ayurveda: Ancient Indian Medical Knowledge for Modern-Day Problems* (Lotus Press, 2001)
8. **David Frawley** – *Yoga and Ayurveda: Self-Healing and Self-Realization* (Lotus Press, 1999)

9. **Melanie Mitchell** – *Artificial Intelligence: A Guide for Thinking Humans* (Farrar, Straus and Giroux, 2019)
10. **Christopher M. Bishop** – *Pattern Recognition and Machine Learning* (Springer, 2006)
11. **Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein** – *Introduction to Algorithms* (MIT Press, 2022)
12. **Jared Dean** – *Big Data, Data Mining, and Machine Learning* (Wiley, 2014)
13. **Thomas Erl** – *Cloud Computing: Concepts, Technology & Architecture* (Prentice Hall, 2013)

Course Name: **Cultural Heritage of Bharat**

Credits: **02**

Course Code: **HIS473**

Paper: **IDC**

Course Objectives: This course aims to introduce the student to the historical and cultural heritage of Bharat and its various types and components.

Course Outcomes: After the completion of the course, the student will be able to:

- Understand the historical and cultural significance of various traditions of the land.
- Know the regional and global extent of the rich cultural heritage of the land.
- Know about the components and types of Bhartiya cultural heritage and the need for their conservation.
- Analyze and demonstrate the ability to construct, analyze and evaluate the trends concerning the traditional heritage of the land.

Evaluation Criteria:

1. Continuous Internal Assessment (CIA): 30%
2. External End Semester Examination (ESE): 70%

Unit-I: **Main Features of Indian Culture and Dharma Systems**

Module/Chapter 1: **Main Features of Indian Culture and Dharma Systems**

Introduction, Characteristics of Indian Culture-Unity in Diversity, Spirituality, Tolerance, Adaptability; Overview of Major Religious Traditions; Religious Traditions-Hindu Dharma: Dharma, Karma & Moksha, Jaina Dharma: Ahimsa & Satya, Bauddha Dharma: Four Noble Truth & Eight-Fold Path; Vedic and Puranic Dharama-Rituals & amp; Hymns, Vedas Sacred texts: Rigveda, Yajurveda, Samaveda, and Atharvaveda.

Module/Chapter 2: **Main Features of Dharma Systems**

Introduction; Vedic and Puranic tradition-Mythology (epic), Moral Brahmanical authority; Varna and Ashram System, Social order and Jati, Individual spiritual growth and responsibilities

Unit-II: **The Role of Tradition in Learning**

Module/Chapter 3: **Samskaras and Education in Ancient India**

Introduction; Samskaras and Purusharthas-The life-cycle rituals in Hindu tradition, Overview of 16 Samskaras (sacraments) and their educational significance, Purusharthas: Dharma, Artha, Kama, and Moksha – their role in shaping Education; Centres of Ancient Indian Education, Overview of ancient learning centers - Nalanda: Structure, Subjects taught, Takshashila: Curriculum and notable scholars, Vikramshila, Vallabhi, and other educational institutions

Module/Chapter 4: **Religious Scriptures and Sant Parampara- I**

Introduction; Introduction to Ramayana – Author (Valmiki), Historical Context, Importance- Key Characters – Rama, Sita, Lakshmana, Hanuman, Ravana, etc, Moral & Ethical Lessons – Dharma, righteousness, devotion Comparative Studies: Valmiki Ramayana vs. Tulsidas' Ramcharitmanas. Regional versions (Kamba Ramayanam, Krittivasi Ramayan), Philosophical Themes: Dharma, Bhakti, Karma. Role of women, kingship, and justice

Module/ Chapter 5: **Religious Scriptures and Sant Parampara- II**

Introduction to Mahabharata – Sage Vyasa, historical importance; Key Characters – Pandavas, Kauravas, Krishna, Bhishma, Draupadi, Historical & Cultural Perspectives: Influence on Indian society, politics, and traditions; Literary Analysis: Poetic style, metaphors, symbolism; Religious and Ethical Studies: How the epics define Hindu beliefs; Contributions of Kalidasa: Shakuntala, Meghaduta, and their impact; Works of Bhasa, Bharavi, and other literary figures; Tulsidas and his contributions to spiritual education; The Sant Parampara: Teachings of Kabir, Mirabai, and their influence on education.

Unit-III: **Indian Culture**

Module/Chapter 6: **Mauryan, Gandhara, and Gupta Art**

Introduction; Mauryan Art- Pillars – Ashokan pillars with inscriptions and animal capitals (e.g., Lion Capital of Sarnath), Stupas – Early Buddhist Stupas like Sanchi and Bharhut, Sculptures – Yaksha-Yakshi figures, stone carvings, and royal representations; Gandhara and Mathura Schools of Art- Gandhara School – Greco-Buddhist influence, depiction of Buddha with wavy hair and drapery, Mathura School – Indigenous style, red sandstone sculptures, life-size Buddha images; Gupta Period's Contribution to Art and Sculpture- Temple Architecture – Development of Nagara style, famous temples (e.g., Dashavatara Temple, Deogarh), Sculptural Excellence – Graceful Buddha images, refined deity sculptures (e.g., Sarnath Buddha). Medieval Architecture

Module/Chapter 7: **South Indian Architecture and Bhakti Movement**

Introduction; Chola Style: Temple Structures – Brihadeshwara Temple, Gangaikonda Cholapuram., Bronze Sculptures – Nataraja, Shiva-Parvati images, Pallava Style: Rock-Cut Temples – Mahabalipuram (Shore Temple, Rathas). Structural Temples – Kailasanatha Temple at Kanchipuram, Vijayanagara Style : Dravidian Influence – Tall gopurams (Virupaksha Temple); Monolithic Sculptures – Ugra Narasimha, Ganesha statues in Hampi; Bhakti Movement-Key Saints and Their Contributions: Alvars and Nayanars – Tamil Vaishnavite and Shaivite saints, North Indian Bhakti Saints – Kabir, Tulsidas, Mirabai, Guru Nanak, Cultural and Artistic Impact of the Movement: Temple Construction and Art – Bhakti-inspired temple murals, devotional sculptures, Literature and Music – Bhajans, kirtans, and literary works like Ramcharitmanas.

Module/Chapter 8: **Introduction to Socio-Religious Reform Movements**

Introduction -Causes of socio-religious reforms, Role of education and modern ideas in reform movements, Brahmo

Samaj, Ideals: Monotheism, social reforms, rejection of idol worship, Arya Samaj, Vedic revivalism and opposition to idol worship, Social reforms: Education, widow remarriage, women's rights, Shuddhi Movement (reconversion to Hinduism); Swami Vivekananda and Key ideas: Practical Vedanta, Universal Religion, Spiritual Nationalism-Role in Indian nationalism and youth empowerment, Establishment of Ramakrishna Mission and its impact, Abindranath Tagore and His Vision, Views on nationalism and humanism, Literary and artistic contributions, Influence on Indian culture and society Sri Aurobindo – The Mystic Revolutionary, Spiritual philosophy and Integral Yoga, Concept of Divine Life and Supermind, Contributions to education and social transformation

Suggested Readings

1. Basham, A.L. *The Wonder That Was India*. Picador Indian Edition, 2004.
2. Singhania, Nitin. *Indian Art and Culture*. Tata Mc Graw Hill: New Delhi, 2017.
3. www.heritageintach.org
4. Biswas, S.S. *Protecting the Cultural Heritage (National Legislations and International Conventions)*. New Delhi: INTACH, 1999.
5. Lahiri, N. *Marshaling the Past - Ancient India and its Modern Histories*. Ranikhet: Permanent Black. 2012, (Chapters 4 and 5).
6. Mathur, Anurag. *Indian Culture and Heritage*. Vol. I. Create Space Independent Publishers, 2017.
7. Vatsayayan, Kapila. *The Cultural Heritage of India*. Vedanta Publishers, 2006
8. www.asi.nic.in
9. www.unesco.org

SEMESTER III

Course Code: **MCA 533**

Name of Course: **Cyber Security**

Course Coordinator: **Dr. Parveen Sadotra**

Credit: **4**

Course Objectives:

- ✓ To provide a fundamental understanding of cyberspace, cybersecurity principles, and the necessity of securing digital systems.
- ✓ To familiarize students with various types of malware, cyber threats, and vulnerabilities, along with their impacts and mitigation strategies.
- ✓ To introduce cryptographic techniques and network security mechanisms for securing digital communication and protecting sensitive information.
- ✓ To explore cyber forensics methodologies, investigative techniques, and legal frameworks for handling cyber crimes and digital evidence.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1 - Demonstrate a thorough understanding of cybersecurity fundamentals, the digital landscape, and the importance of protecting digital assets.

CO^2 - Identify, analyze, and mitigate different types of cyber threats, including malware, phishing, and social engineering attacks.

CO^3 - Apply cryptographic techniques and network security measures to enhance the security of information systems and digital communication.

CO^4 - Conduct cyber forensic investigations, analyze digital evidence, and understand the legal frameworks governing cyber crimes and security practices.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content

UNIT-I: Cybersecurity Fundamentals

Chapter/Module 1: Introduction to Cyberspace and Cybersecurity

Definition and evolution of Cyberspace: Understanding the digital world, its components, and interconnectivity, **Importance and scope of Cybersecurity:** Protecting digital assets, ensuring data integrity, and mitigating cyber threats, **The need for cybersecurity in modern digital systems:** Rising cyber threats, impact on businesses, and individual data protection.

Chapter/Module 2: Types of Malware

Introduction to malicious software (Malware): Understanding the intent and impact of malicious software, **Types of Malware:** Worms: Self-replicating programs that spread without user intervention, Viruses: Malicious code that attaches to files and spreads upon execution, Spyware: Software that secretly gathers user information, Trojans: Malicious software disguised as legitimate programs, **Impact of Malware on systems and networks:** Data theft, system corruption, financial loss, and privacy breaches.

Chapter/Module 3: Cyber Security Breaches and Vulnerabilities

Phishing: Definition, types, and prevention methods (Types: Spear phishing, whaling, vishing, and smishing, (Prevention: Email filtering, user education, and security awareness)
Identity Theft: Techniques used and protective measures, Social engineering, data breaches, and credential theft, Preventive steps: Strong passwords, multi-factor authentication, credit monitoring
Harassment in cyberspace: Cyberbullying, stalking, and mitigation, Types of cyber harassment: Doxing, trolling, revenge porn, Legal actions and preventive measures
Introduction to system vulnerabilities and exploitation: Software vulnerabilities, unpatched systems, and insider threats, Importance of timely updates and vulnerability assessments.

UNIT-II: Cyber Threats and Prevention

Chapter/Module 4: Types of Cyber Attacks

Password Attacks: Brute force, dictionary attacks, credential stuffing, Denial of Service (DoS) and Distributed DoS (DDoS) attacks: Mechanisms, botnets, and mitigation techniques, Passive Attacks: Eavesdropping, traffic analysis, man-in-the-middle attacks, Advanced threats: Zero-day and zero-click attacks: How they exploit system vulnerabilities and their impact, Basics of Penetration Testing: Ethical hacking, vulnerability scanning, and security assessments.

Chapter/Module 5: Prevention and Protection Strategies

Crafting a strong password and best practices: Using password managers, passphrases, and avoiding common pitfalls, **Two-Step Verification and authentication techniques:** OTP-based, biometric, and hardware authentication methods, **Safe downloading practices and malware prevention:** Avoiding pirated software, scanning downloads, and sandboxing, **Evaluating website legitimacy to prevent cyber threats:** Checking SSL certificates, avoiding phishing links, and using secure connections.

Chapter/Module 6: Legal Perspective on Cyber Crimes

Overview of the IT Act 2000 and its amendments: Key provisions, cybercrime classifications, and penalties, **Key Cyber Crime Offences under Indian Law:** Data theft, identity fraud, cyberstalking, hacking, and financial fraud, **Legal remedies and**

penalties for cyber crimes: Filing complaints, cyber cells, penalties, and jurisdiction challenges.

UNIT-III: Cryptography and Network Security

Chapter/Module 7: Introduction to Cryptography

Basic principles of cryptography: Confidentiality, integrity, authentication, and non-repudiation, **Importance of encryption and decryption in cybersecurity:** Securing communication, protecting sensitive data, and compliance requirements.

Chapter/Module 8: Cryptographic Techniques

Symmetric Key Cryptography: Algorithms and applications (DES, AES, Blowfish, and their security implications), **Asymmetric Key Cryptography:** Public and private keys (RSA, ECC, Diffie-Hellman key exchange), **Real-world applications of cryptographic techniques:** Secure messaging, digital banking, and blockchain security

Chapter/Module 9: Authentication and Digital Security

Message Authentication techniques: Hash functions (SHA, MD5), HMAC, and integrity checks, **Digital Signatures and their role in security:** Ensuring authenticity, non-repudiation, and legal validity, **Cryptographic protocols for secure communication:** SSL/TLS, IPsec, PGP, and their role in cybersecurity.

Chapter/Module 10: Firewalls and Network Security

Overview of Firewalls: Importance and functionality (Packet filtering, proxy-based, and stateful inspection) **Types of Firewalls:** Hardware vs. software, next-generation firewalls, **Role of firewalls in securing network infrastructure:** Preventing unauthorized access, monitoring network traffic, and intrusion detection.

UNIT-IV: Cyber Forensics and Investigations

Chapter/Module 11: Introduction to Cyber Forensics

Definition and scope of Cyber Forensics: Investigating cyber crimes, collecting digital evidence, **Importance of digital forensic investigations:** Role in legal proceedings, corporate security, and law enforcement

Chapter/Module 12: Conducting Cyber Investigations

Handling preliminary investigations: Identifying sources, securing digital evidence, **Controlling and managing forensic investigations:** Chain of custody, forensic methodologies, and compliance, **Disk-based analysis techniques:** Recovering deleted files, file system analysis, imaging tools.

Chapter/Module 13: Advanced Cyber Forensics Techniques

Investigating Information-hiding techniques: Steganography detection, data obfuscation analysis, **Email Forensics:** Scrutinizing email headers and metadata (Identifying spoofed emails, tracking IP addresses), **Internet activity tracing and real-time memory tracking:** Browser forensics, session reconstruction, RAM analysis.

Chapter/Module 14: Cyber Crime and Security Organizations

Overview of organizations addressing cyber crimes in India (CERT-In, NCIIPC, Cybercrime Investigation Cells), **Role of law enforcement agencies and CERT-In:** Cyber threat intelligence, national security coordination, International cooperation in cyber crime mitigation: INTERPOL, Europol, and global cybersecurity initiatives

Suggested Readings:

1. Zach Webber, “Cyber Security”.

Course Code: **MCA 601**

Name of Course: **Research Methodology**

Course Coordinator: **Dr. Pradeep Chouksey**
Credit: **4**

Course Objectives:

- ✓ To introduce fundamental concepts of research, including its objectives, significance, and applications across various disciplines.
- ✓ To familiarize students with different research types, approaches, and methodologies for systematic investigation.
- ✓ To develop skills in conducting literature reviews, identifying research gaps, and formulating research problems effectively.
- ✓ To equip students with the ability to define research objectives, establish operational definitions, and construct valid hypotheses.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1 - Demonstrate a clear understanding of research as a structured process and its importance in scientific and academic advancements.

CO^2 - Identify and differentiate various research types, approaches, and methodologies based on specific research problems.

CO^3 - Conduct a comprehensive literature review to assess existing research, identify gaps, and formulate well-defined research objectives.

CO^4 - Develop research hypotheses and variables using appropriate frameworks while ensuring clarity and specificity in research objectives.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content

UNIT - I: Introduction & Research Problem Formulation

Chapter/Module 1: Research Objectives and Motivation

Definition of Research: Understanding research as a systematic investigation to establish facts and reach new conclusions, **Research in Different Fields**: Overview of research applications in science, technology, social sciences, and humanities, **Importance of Research**: How research drives innovation, policy-making, and

technological advancements,**Defining Research Objectives**: Characteristics of well-defined research objectives, **Motivations Behind Conducting Research**: Academic curiosity, problem-solving, societal impact, business insights, and career advancement.

Chapter/Module 2: Types of Research and Research Approaches

Classification of Research: Basic (pure) vs. applied research, exploratory vs. descriptive, longitudinal vs. cross-sectional, **Qualitative Research**: Case studies, ethnography, content analysis, interviews, **Quantitative Research**: Experiments, surveys, statistical analysis, hypothesis testing, **Mixed-Methods Research**: Integrating qualitative and quantitative approaches for comprehensive insights, **Choosing the Right Approach**: Factors affecting research approach selection based on research problem and data availability.

Chapter/Module 3: Research Method vs. Methodology

Understanding Research Methodology: The systematic framework guiding research, **Difference Between Method and Methodology**: Methods as specific techniques, methodology as an overarching approach, **Common Research Methods**: Surveys, case studies, experimental methods, content analysis, **Developing a Methodological Framework**: Selecting appropriate methods based on research questions

Chapter/Module 4: Research Process and Literature Review

Steps in the Research Process: Identifying problems, reviewing literature, designing research, collecting data, analyzing findings, **Conducting a Literature Review**: Sources of literature, academic databases, citation management tools, **Identifying Research Gaps**: How to spot unexplored areas for potential research contributions, **Critical Analysis of Literature**: Techniques for evaluating research credibility and relevance.

Chapter/Module 5: Formulation of Objectives, Definitions, and Hypotheses

Setting Research Objectives: SMART (Specific, Measurable, Achievable, Relevant, Time-bound) framework, **Establishing Operational Definitions**: Defining key terms

for clarity and consistency, **Identifying Variables**: Independent, dependent, confounding, and control variables, **Constructing Hypotheses**: Types (null vs. alternative), characteristics, and formulation techniques.

UNIT - II: Research Design & Data Collection

Chapter/Module 6: Research Design – Need and Characteristics

Importance of Research Design: Ensuring validity, reliability, and accuracy, **Key Characteristics**: Objectivity, precision, flexibility, and ethical considerations, **Choosing the Right Research Design**: Aligning with research goals and available resources.

Chapter/Module 7: Types of Research Design

Descriptive Research: Observational studies, case studies, **Experimental Research**: Controlled experiments, randomized trials, **Exploratory Research**: Pilot studies, qualitative insights, **Longitudinal vs. Cross-Sectional Studies**: Differences and applications

Chapter/Module 8: Principles of Experimental Research Design

Key Principles: Control, randomization, replication, and generalizability, **Threats to Internal and External Validity**: Confounding variables, selection bias, measurement errors, **Types of Experimental Designs**: Between-subjects, within-subjects, factorial designs.

Chapter/Module 9: Methods of Data Collection

Primary Data Collection: Surveys, interviews, focus groups, direct observation, **Secondary Data Collection**: Literature reviews, government reports, archival data, **Choosing the Right Data Collection Method**: Factors influencing selection

Chapter/Module 10: Ethical Issues in Data Collection

Ethical Principles: Integrity, objectivity, confidentiality, **Informed Consent**: Importance and implementation in research, **Data Privacy and Protection**: Ensuring

compliance with regulations like GDPR, HIPAA, **Plagiarism and Research Integrity**: Avoiding unethical research practices.

UNIT - III: Sampling & Data Analysis

Chapter/Module 11: Sampling and Statistical Distributions

Importance of Sampling: Reducing research costs while maintaining validity, **Types of Sampling Techniques**: Probability (random, stratified, cluster) vs. non-probability (convenience, snowball), **Central Limit Theorem**: Importance in research, **Sampling Distributions**: Understanding the normal distribution and other probability models.

Chapter/Module 12: Data Analysis and Interpretation

Descriptive Statistics: Measures of central tendency (mean, median, mode), **Measures of Dispersion**: Variance, standard deviation, range, **Correlation vs. Causation**: Understanding relationships between variables, **Regression Analysis**: Predicting relationships between dependent and independent variables.

Chapter/Module 13: Data Visualization and Presentation

Importance of Data Visualization: Enhancing data interpretation, **Types of Graphs and Charts**: Line charts, bar graphs, scatter plots, **Best Practices in Data Presentation**: Avoiding misrepresentation.

UNIT - IV: Hypothesis Testing & Research Proposal Writing

Chapter/Module 14: Hypothesis Testing and Research Proposal Writing

Understanding Hypothesis Testing: Defining null and alternative hypotheses, **Types of Hypothesis Tests**: Parametric (t-tests, ANOVA) vs. non-parametric (chi-square, Wilcoxon tests), **Interpreting p-Values and Confidence Intervals**: Decision-making in research, **Limitations of Hypothesis Testing**: False positives/negatives, sample size concerns, **Chi-Square Test and ANOVA**: Applications and interpretation, **Writing a Research Proposal**: Structure, key components (background, objectives, methodology), **Common Mistakes in Research Proposals**: How to avoid them.

Textbooks:

1. **C.R. Kothari & Gaurav Garg** – *Research Methodology: Methods and Techniques* (New Age International, 4th Edition, 2019)
2. **Ranjit Kumar** – *Research Methodology: A Step-by-Step Guide for Beginners* (SAGE Publications, 5th Edition, 2019)
3. **John W. Creswell & J. David Creswell** – *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (SAGE Publications, 5th Edition, 2018)
4. **K. N. Krishnaswamy, Appa Iyer Sivakumar & M. Mathirajan** – *Management Research Methodology: Integration of Principles, Methods and Techniques* (Pearson Education, 2010)

Reference Books:

5. **Alan Bryman** – *Social Research Methods* (Oxford University Press, 5th Edition, 2015)
6. **Uma Sekaran & Roger Bougie** – *Research Methods for Business: A Skill-Building Approach* (Wiley, 7th Edition, 2016)
7. **Zikmund, Babin, Carr & Griffin** – *Business Research Methods* (Cengage Learning, 9th Edition, 2016)
8. **Robert K. Yin** – *Case Study Research and Applications: Design and Methods* (SAGE Publications, 6th Edition, 2018)
9. **Joseph F. Hair, Jr., Arthur H. Money, Phillip Samouel, & Michael Page** – *Essentials of Business Research Methods* (Routledge, 3rd Edition, 2019)
10. **Neuman, W. Lawrence** – *Social Research Methods: Qualitative and Quantitative Approaches* (Pearson, 7th Edition, 2014)

Course Code: **MCA 516**
Name of Course: **Web Programming**
Course Coordinator: **Mr. Mayank Chopra**

Credit: **4**

Course Objectives:

- To introduce students to the fundamentals of PHP programming and its integration with HTML.
- To develop problem-solving abilities using PHP control structures, functions, and arrays.
- To familiarize students with object-oriented programming concepts, form handling, and file management in PHP.
- To provide hands-on experience in image manipulation, database connectivity, and data management using PHP.

Course Outcomes: After the successful completion of this course, the student will be able to

CO[1]- Demonstrate a clear understanding of PHP programming fundamentals, including syntax, variables, data types, and control structures.

CO2- Apply PHP functions, arrays, and string handling techniques to build dynamic web applications**.**

CO[3]- Implement object-oriented concepts, manage HTML forms, and perform file handling operations using PHP.

CO[4]- Create and manipulate images using the GD library, connect to MySQL databases, and perform CRUD operations for effective data management**.**

Evaluation Criteria:
A student will be evaluated on the following basis:
End Semester Examination: 70% weightage
Continuous Internal Assessment: 30% weightage

Unit-I: Introduction to PHP and Control Structures
Module/ Chapter 1: Introduction to PHP
How PHP Scripts Work, PHP Syntax and Writing First PHP Program, Embedding PHP in HTML and HTML in PHP
Module/ Chapter 2: PHP Basics
PHP Data Types, Variables in PHP, Operators in PHP
Module/ Chapter 3: Control Structures
Conditional Statements:if, if-else, if-elseif-else, Nested if, Switch Statement
Module/ Chapter 4: Looping Structures
For Loop, While Loop, Do-While Loop, For Each Loop Function

Unit-II: Functions, Arrays, and String Handling
Module/ Chapter 5: Functions in PHP

Introduction and Syntax, User-Defined Functions and System-Defined Functions, Parameterized Functions, Date & Time Functions, Hash Functions, Mail Functions

Module/ Chapter 6: Arrays in PHP

Array Syntax and Declaration, Associative Arrays, Numeric Arrays, Multidimensional Arrays

Module/ Chapter 7: String Handling

Creating and Accessing Strings, Searching, Replacing, and Formatting Strings, String-Related Library Functions, Introduction to Regular Expressions, Pattern Matching, Replacing Text, and Splitting Strings with Regular Expressions

Unit-III: Object-Oriented Programming, Forms, and File Handling

Module/ Chapter 8: Object-Oriented Programming in PHP

Creating Classes and Objects, Working with Methods and Properties, Overloading and Inheritance, Constructors and Destructors

Module/ Chapter 9: Handling HTML Forms with PHP

Form Submission and Processing, Preserving State with Query Strings, Cookies, and Sessions

Module/ Chapter 10: File Handling in PHP

Understanding Files and Directories, D Opening, Closing, Reading, Writing, Copying, Renaming, and Deleting Files, Working with Directories, Building a Simple Text Editor in PHP

Unit-IV: Turing Machines & Computational Complexity

Module/ Chapter 11: Basics of Computer Graphics

Introduction to Computer Graphics in PHP, Understanding Image Formats and Colors

Module/ Chapter 12: Creating and Manipulating Images

Creating Images using PHP GD Library, Manipulating Images: Resizing, Cropping, and Adding Effects, Adding Text to Images

Module/ Chapter 13: Database Connectivity with MySQL

Connecting PHP to MySQL Database, Executing SQL Queries using PHP

Module/ Chapter 14: Data Manipulation with PHP

Retrieving and Displaying Data from MySQL, Inserting, Updating, and Deleting Data using PHP, Error Handling in Database Operations

Suggested Readings:

- Matt Doyale, “Beginning PHP 5.3”, Wiley India Edition.
- Larry Ulman, “PHP and MySQL5”, Pearson.
- Robert Sebesta,” Programming with World Wide Web”, Pearson.
- John Duckett, “Beginning with HTML, XHTML, CSS and Javascript”, Wiley-Wrox
- Building PHP Framework Applications WithSymfony, CakePHP, AndZend, by BartoszPorebski Karol PrzystalskiLeszek Nowak, Wiley India.

Course Code: MCA 604

Name of Course: Review of Literature

Course Coordinator: **Mr. Mayank Chopra**
Credit: **4**

Course Objectives

- Develop the ability to critically analyse academic literature in the field of Computer Science.
- Enhance research and analytical skills by identifying relevant research papers, journals, and articles.
- Formulate research problems and justify their significance through a systematic review of literature.
- Improve academic writing and referencing skills using standard citation formats.
- Prepare a comprehensive literature review report for future research work or thesis.

Course Outcomes: After successful completion of this course, learners will be able to:

CO[1]- Identify credible sources of information related to their research area.
CO[2]- Analyse and synthesize information from multiple academic papers.
CO[3]- Identify research gaps and suggest potential research directions.
CO[4]- Develop a structured and well-referenced literature review report.
CO[5]- Demonstrate understanding of ethical practices in academic research.

1. Evaluation Breakdown

Mode of Evaluation	Weightage	Marks	Component
Internal Evaluation	30%	60	Continuous Internal Assessment*
External Evaluation	70%	140	• Review Article Evaluation • Viva Voce
Total	100%	200	-

**The allotted mentors will conducting the internal evaluation (30%) under Continuous Internal Assessment.*

2. Literature Review Guidelines

2.1. Topic Selection & Approval

- Students must select a topic in consultation with the course coordinator.
- Approval must be obtained via email/official online submission.
- Topics should align with current industry trends, academic relevance, or societal impact.
- Work must be completed independently (no financial support from the department); Students can seek external funding.
- Students may seek guidance from concerned course coordinator but must ensure originality.
- Timeline: Strict adherence to department deadlines (late submissions may incur penalties)

2.2. Literature Review Documentation Requirements

The Literature Review article (6-20 pages) must demonstrate original work with specific focus on topic.

Structure of Literature Review Article

1. **Title Page**: Includes the title of the literature review, student's name, enrolment number, course name, instructor's name, and submission date.
2. **Abstract:** A summary (150-250 words) that highlights the research focus, key findings, and conclusions.
3. **Keywords:** 4-6
4. **Introduction:**
    - Provide context for the research topic.
    - Explain the significance and purpose of the literature review.
    - State the research questions or objectives.
5. **Methodology:**
    - Describe the search strategy used for selecting relevant articles (e.g., databases searched, keywords used).
    - Outline inclusion and exclusion criteria.
    - Justify the choice of sources.
6. **Thematic or Chronological Review:**
    - Organize the literature based on themes, concepts, or trends.
    - Alternatively, present the literature in a chronological order to show the development of research over time.
7. **Results and Findings:**
    - Evaluate the strengths and limitations of previous research.
    - Identify gaps, contradictions, and emerging trends.
8. **Discussion and Synthesis:**
    - Compare and contrast various perspectives.
    - Propose potential areas for further research.
9. **Conclusion:**
    - Summarize key findings.
    - Emphasize the contributions of the reviewed literature to the field.
10. **References:**
    - List all cited sources using a consistent citation style IEEE.
11. **Appendices (If Applicable):**
    - Include supplementary materials like charts, graphs, or tables.

2.3. Formatting Standards

2.3.1. Paper Format

- **Paper Size**: A4 (210mm x 297mm)
- **Margins:**
    - Top: 19mm (0.75 inches)
    - Bottom: 19mm (0.75 inches)
    - Left and Right: 13mm (0.5 inches)
- **Font and Text**

- **Font Type**: Times New Roman
- **Font Size**:
  - Title: 24 pt, centred, bold
  - Author Names and Affiliations: 10 pt, centred
  - Abstract: 9 pt, justified
  - Keywords: 9 pt, italicized
  - Body Text: 10 pt, justified.
  - Section Headings: 10 pt, bold, capitalized.
  - Subsection Headings: 10 pt, italicized.
  - References: 8 pt

2.3.2. Title and Author Details

- The title should be in bold with no abbreviations.
- List all authors with their full names.
- Include affiliations, email addresses, and locations.

2.3.3. Body of the Paper

- Divide the paper into sections such as Introduction, Methodology, Results, Discussion, and Conclusion.
- Use numbered headings (e.g., 1. Introduction, 2. Methodology).
- Figures and tables should be placed within the text, centred, and numbered with descriptive captions.
- Equations should be numbered sequentially and aligned to the right using equation **editor tools.**

2.3.4. Citations and References

- **Use IEEE citation style.**
- In-text citation example: [1]

3. Evaluation Parameters of review article (Maximum marks 70)

Evaluation Criteria	**Weightage**
Clarity and Definition of Research Problem	20%
Review of Literature	25%
Research Objectives and Hypotheses	15%
Methodology	25%
Ethical Considerations and Timeline	15%

4. Viva-Voce Guidelines

4.1. Evaluation Process

Duration: 30 minutes (15 min presentation + 15 min Q&A)
Platform: As Decided by CDOE, CUHP

4.2. Evaluation Criteria (Maximum marks 70)

Evaluation Criteria	**Weightage**
Understanding of Research Problem and Objectives	20%
Justification of Methodology	25%

Knowledge of Relevant Literature	20%
Clarity and Presentation Skills	15%
Ability to Defend and Respond to Questions	20%

5. Plagiarism Policy

As per UGC (Promotion of Academic Integrity and Prevention of Plagiarism in Higher Educational Institutions) Regulations, 2018, plagiarism is strictly prohibited.

Plagiarism Levels & Penalties

Similarity Percentage	Consequences
Up to 10%	No penalty
10% – 40%	Revision required within a stipulated time
40% – 60%	Resubmission after a 6-month cooling period
Above 60%	Cancellation of dissertation and disciplinary action

6. Final Assessment

The overall score will be based on the successful completion of both the dissertation and viva-voce while ensuring compliance with UGC plagiarism norms.

7. Suggested Readings

- Kumar, R., *Research Methodology: A Step-by-Step Guide for Beginners*, SAGE Publications, 5th Edition, 2021.
- Kothari, C. R., *Research Methodology: Methods and Techniques*, New Age International Publishers, 4th Edition, 2022.
- Creswell, J. W., & Creswell, J. D., *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*, SAGE Publications, 5th Edition, 2018.
- Machi, L. A., & McEvoy, B. T., *The Literature Review: Six Steps to Success*, Corwin Press, 4th Edition, 2022.
- Hart, C., *Doing a Literature Review: Releasing the Social Science Research Imagination*, SAGE Publications, 2018.
- Zobel, J., *Writing for Computer Science*, Springer, 3rd Edition, 2014.
- Dawson, C. W., *Projects in Computing and Information Systems: A Student's Guide*, Pearson, 2nd Edition, 2015.
- Glass, R. L., *Facts and Fallacies of Software Engineering*, Addison-Wesley, 2003.
- van Vliet, H., *Software Engineering: Principles and Practice*, Wiley, 3rd Edition, 2008.
- Webster, J., & Watson, R. T., *Analyzing the Past to Prepare for the Future: Writing a Literature Review*, MIS Quarterly, Vol. 26, No. 2, 2002.
- Kitchenham, B., & Charters, S., *Guidelines for Performing Systematic Literature Reviews in Software Engineering*, EBSE Technical Report, 2007.
- IEEE, *IEEE Citation Guide*, IEEE, 2021.

- American Psychological Association, *Publication Manual of the American Psychological Association*, APA, 7th Edition, 2020.
- Modern Language Association, *MLA Handbook*, MLA, 9th Edition, 2021.

Course Code: **MCA 605**

Name of Course: **Research Proposal**
Course Coordinator: **Sh. Manoj Dhiman**
Credit: **4**

Course Objectives

- Develop the ability to conceptualize, structure, and articulate a comprehensive research proposal.
- Enhance analytical thinking by identifying relevant research problems and proposing viable research questions.
- Strengthen the understanding of existing literature and apply it to justify research objectives.
- Formulate appropriate research methodologies, including data collection and analysis techniques.
- Demonstrate clear academic writing, proper referencing, and persuasive research argumentation.

Course Outcomes: After successful completion of this course, learners will be able to:

CO1: Develop a clear and coherent understanding of research processes.
CO2: Conduct a comprehensive literature review to justify the research problem.
CO3: Design a robust methodology appropriate to the proposed research objectives.
CO4: Write and present a well-structured academic research proposal adhering to academic standards.
CO5: Defend and critically discuss their research proposal during the viva-voce.

1. Evaluation Breakdown

Mode of Evaluation	Weightage	Marks	Component
Internal Evaluation	30%	60	Continuous Internal Assessment*
External Evaluation	70%	140	• Research Proposal Evaluation • Viva Voce
Total	100%	200	-

**The allotted mentors will conducting the internal evaluation (30%) under Continuous Internal Assessment.*

2. Research Proposal Guidelines

2.1. Topic Selection & Approval

- Students must select a research topic with the approval of the course coordinator.
- The topic should be current, relevant to the field of Computer Applications, and academically significant.

- Independent research is encouraged, while external support can be acknowledged.
- Timely submission of the approved topic is mandatory.

2.2. Proposal Structure

The proposal should include the following sections:

1. **Title Page:**
   - Title
   - Student Name
   - Enrolment Number
   - Course Name
   - Instructor's Name
   - Date of Submission
2. **Abstract:**
   - 150-250 word summary of the proposal.
3. **Introduction:**
   - Introduction to the research problem, its significance, and objectives.
4. **Literature Review:**
   - Comprehensive review of existing work, identifying gaps and justifying the proposed study.
5. **Proposed Methodology:**
   - Explanation of research design, methods of data collection and analysis.
6. **Expected Outcomes:**
   - Description of potential results and contributions of the research.
7. **Timeline:**
   - Clear work plan and estimated timeline for research completion.
8. **References:**
   - Proper citation of all references in IEEE style.

3. Evaluation Parameters for Research Proposal

Evaluation Criteria	Weightage
Clarity and Definition of Research Problem	20%
Review of Literature	25%
Research Objectives and Hypotheses	15%
Methodology	25%
Ethical Considerations and Timeline	15%

4. Viva-Voce Guidelines

4.1. Evaluation Process

- **Duration:** 30 minutes (15 min presentation + 15 min Q&A)
- **Platform:** As Decided by CDOE, CUHP

4.2. Evaluation Criteria

Evaluation Criteria	Weightage
Understanding of Research Problem and Objectives	20%

Justification of Methodology	25%
Knowledge of Relevant Literature	20%
Innovation and Originality of Proposed Work	30%
Clarity and Presentation Skills	15%
Ability to Defend and Respond to Questions	20%

5. Plagiarism Policy

As per UGC (Promotion of Academic Integrity and Prevention of Plagiarism in Higher Educational Institutions) Regulations, 2018, plagiarism is strictly prohibited.
Plagiarism Levels & Penalties

Similarity Percentage	Consequences
Up to 10%	No penalty
10% – 40%	Revision required within a stipulated time
40% – 60%	Resubmission after a 6-month cooling period
Above 60%	Cancellation of dissertation and disciplinary action

6. Final Assessment

The overall score will be based on the successful completion of both the dissertation and viva-voce while ensuring compliance with UGC plagiarism norms.

7. Suggested Readings

- Kothari, C. R. (2004). *Research Methodology: Methods and Techniques*. New Age International.
- Creswell, J. W. (2018). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. Sage publications.
- Singh, Y. K. (2006). *Fundamentals of Research Methodology and Statistics*. New Age International.
- Walliman, N. (2011). *Research Methods: The Basics*. Routledge.

SEMESTER IV

Course Code: **MCA 519**

Name of Course: **Computer Networks**

Course Coordinator: **Dr. Parveen Sadotra**
Credit: **4**

Course Objectives:

- ✓ To introduce fundamental concepts of computer networks, their components, and applications across various domains.
- ✓ To provide an in-depth understanding of network models, including OSI and TCP/IP, and their significance in communication.
- ✓ To explore key networking protocols, switching techniques, and transmission media used in data communication.
- ✓ To familiarize students with data link, network, and transport layer functionalities, including error control, routing, and congestion control mechanisms.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1 - **Understand** the basics of computer networks, their types, and the role of various components in facilitating communication.

CO^2 - **Analyze** the OSI and TCP/IP models, explaining the functions of each layer and their real-world applications.

CO^3 - **Evaluate** different network topologies, switching techniques, and transmission media for efficient data transfer.

CO^4 - **Demonstrate** knowledge of data link and transport layer protocols, implementing error detection, flow control, and routing algorithms in networking environments.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content

UNIT - I: Introduction to Computer Networks

Chapter/Module 1: Definition of a Computer Network

Overview of computer networks: Definition, purpose, and components**, Importance and applications of networks in various domains** (business, education, healthcare, entertainment), **Types of networks (LAN, MAN, WAN):** Characteristics, advantages, and use cases

Chapter/Module 2: The OSI Reference Model

Introduction to OSI model: Concept, history, and importance, **Seven layers of OSI model:** Physical, Data Link, Network, Transport, Session, Presentation, Application, Functions of each layer with real-world examples, **Protocols and hardware involved in each layer:** Role of routers, switches, bridges, and hubs

Chapter/Module 3: The TCP/IP Reference Model

Overview of TCP/IP model: History, development, and relevance in modern networking, **Layers of TCP/IP model and their functions:** Network Interface, Internet, Transport, and Application, **Comparison of OSI and TCP/IP models:** Key differences, advantages, and limitations

Chapter/Module 4: Application Layer

Domain Name System (DNS) and its role: Structure, hierarchy, working mechanism, **Internet services:** Email protocols (SMTP, POP, IMAP), File Transfer Protocol (FTP) and its working, **World Wide Web (WWW) and Hypertext Transfer Protocol (HTTP):** How webpages are accessed and rendered, **Simple Network Management Protocol (SNMP):** Basics, components, and uses, **Multimedia applications:** Streaming, real-time communication, **Basics of network security:** Threats, encryption, authentication

UNIT - II: Physical Layer

Chapter/Module 5: Network Topologies

Introduction to network topologies: Definition and significance, **Types:** Linear Bus, Ring, Star, Hierarchical/Tree: Structure, working, pros and cons, Topology comparison and considerations for implementation

Chapter/Module 6: Switching Mechanisms

Circuit switching: Concept, working mechanism, advantages, and disadvantages, **Message switching:** Overview, comparison with circuit switching, **Packet switching:** Explanation, types (datagram & virtual circuit), benefits, and drawbacks, Advantages and disadvantages of each switching technique

Chapter/Module 7: Transmission Media

Guided media: Twisted pair (STP, UTP), Coaxial cable, Optical fiber; characteristics, applications, **Unguided media:** Electromagnetic spectrum, Radio transmission,

Microwave transmission, **Comparison of fiber optics and copper wire:** Performance, cost, durability

UNIT - III: Data Link Layer

Chapter/Module 8: Introduction and Design Issues of Data Link Layer

Role of Data Link Layer (DLL) in network communication, **Services provided to the network layer:** Addressing, error control, flow control, **Framing methods and types:** Character count, byte stuffing, bit stuffing

Chapter/Module 9: Error Control and Flow Control

Types of errors in data communication: Single-bit, burst errors, **Error control mechanisms:** Parity bits, Cyclic Redundancy Check (CRC), **Flow control techniques:** Stop-and-Wait, Sliding Window, Go-Back-N, Selective Repeat

Chapter/Module 10: Data Link Layer Protocols

Transmission control protocols: Definition, working principles, **High-Level Data Link Control (HDLC):** Frame structure, modes, error detection

UNIT – IV: Network Layer and Transport Layer

Chapter/Module 11: Design Issues of Network Layer

Nature of services provided by the network layer: Connectionless vs. connection-oriented, **Principles of Internetworking:** Addressing, packet forwarding, fragmentation

Chapter/Module 12: Routing Algorithms

Types of routing algorithms and properties: Static vs. dynamic routing, **Optimality principle, Shortest path algorithm:** Bellman-Ford, Dijkstra, **Flooding, Distance vector routing, Hierarchical routing, Link-state routing:** Explanation and working principles

Chapter/Module 13: Congestion Control

Factors affecting congestion: Network load, bandwidth limitations, **Comparison of flow control and congestion control:** Key differences and use cases, **General principles of congestion control:** Open loop vs. closed loop, **Closed loop solutions:** Congestion detection and avoidance techniques, **Internet Protocol (IPv4):** Addressing, header structure, subnetting

Chapter/Module 14: Transport Layer Protocols

Introduction to Transport Layer services: End-to-end communication, reliability, **Connection establishment and release:** Three-way handshake, FIN-ACK, **Transport layer protocols:** Transmission Control Protocol (TCP) - Reliable, connection-oriented; User Datagram Protocol (UDP) - Connectionless, low-latency

Text Books:

1. Behrouz A. Forouzan, “Data Communications and Networking”, 4E , 2017, Tata McGraw Hill.

Reference Books:

1. Natalia Olifer& Victor Olifer, “Computer Networks”, John Wiley & Sons Ltd., 2013.
2. William Stallings, “Data & Computer Communication”, Pearson Education, 2014.
3. Andrew S. Tanenbaum, “Computer Networks”, Pearson Education, 2017

Course Code: **MCA 602**
Name of Course: **Academic Writing**

Course Coordinator: **Dr. Parveen Sadotra**
Credit: **2**

Course Objectives:

- ✓ To introduce students to the fundamentals of LaTeX, its history, merits, and applications in technical writing.
- ✓ To familiarize students with the installation, compilation process, and structural elements of a LaTeX document.
- ✓ To develop skills in document structuring, including sectioning, layout management, figure handling, and referencing.
- ✓ To equip students with advanced LaTeX features such as mathematical typesetting, bibliography management, and customization using packages.

Course Outcomes: After the successful completion of this course, the student will be able to

CO^1 - Demonstrate a clear understanding of LaTeX and its advantages over traditional word processors for technical and scientific documentation.

CO^2 - Install and configure LaTeX software, compile documents, and structure content effectively using preamble and syntax rules.

CO^3 - Utilize LaTeX for professional document preparation, including figures, tables, mathematical equations, and citations.

CO^4 - Apply LaTeX for real-world applications such as writing research papers, resumes, reports, and presentation slides using appropriate document classes.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content

UNIT - I: Introduction to LaTeX

Chapter/Module 1: History and Basics of LaTeX

A Brief History of LaTeX, What is LaTeX?, Merits of LaTeX over Word Processors, Demerits of LaTeX

Chapter/Module 2: Installation and Compilation

Installation of LaTeX software, Understanding LaTeX Compilation Process, Structure of a LaTeX Input File

Chapter/Module 3: Preamble and Basic Syntax

Introduction to the Preamble, Creating a Title Page, Page Numbering and Headings, Modifying Text, Using LaTeX Packages

UNIT - II: Document Structuring in LaTeX

Chapter/Module 4: Page Layout and Sectioning

Titles and Abstract, Chapters, Sections, and Subsections, List Making Environments, Table of Contents

Chapter/Module 5: Handling Figures and Tables

Figure Handling and Numbering, List of Figures, List of Tables, Creating Tables in LaTeX, Matrix and List Environments

Chapter/Module 6: Index and References

Generating an Index, Cross-Referencing Sections and Figures

UNIT - III: Advanced Features in LaTeX

Chapter/Module 7: Mathematical Typesetting

Writing Mathematical Equations in LaTeX, Using Mathematical Symbols, Generating Mathematical Expressions

Chapter/Module 8: Packages and Customization

Introduction to Packages: Geometry, Hyperref, amsmath, amssymb, Algorithm Writing with Algorithms and Algorithmic Packages, Graphics and Color Handling, Code Listings using Listings Package, Creating Custom Commands

Chapter/Module 9: Bibliography and Citations

Generating Bibliography, Citation Styles and Management, Index Generation

UNIT - IV: Document Classes and Applications

Chapter/Module 10: LaTeX Classes and Applications

Document Classes: article, book, report, beamer, IEEEtran, Writing a Resume in LaTeX, Creating Question Papers, Writing Research Papers and Articles, Preparing Presentation Slides with Beamer

Textbooks:

1. **Leslie Lamport**, *LaTeX: A Document Preparation System*, 2nd Edition, Addison-Wesley, 1994.
2. **Helmut Kopka and Patrick W. Daly**, *A Guide to LaTeX*, 4th Edition, Addison-Wesley, 2003.
3. **Tobias Oetiker et al.**, *The Not So Short Introduction to LaTeX 2ε*, Available online at: https://tobi.oetiker.ch/lshort/
4. **Stefan Kottwitz**, *LaTeX Beginner's Guide*, 2nd Edition, Packt Publishing, 2021.

Reference Books:

1. **George Grätzer**, *More Math into LaTeX*, 5th Edition, Springer, 2016.
2. **Michel Goossens, Frank Mittelbach, and Alexander Samarin**, *The LaTeX Companion*, 2nd Edition, Addison-Wesley, 2004.
3. **Herbert Voß**, *PSTricks: Graphics and PostScript for TeX and LaTeX*, Addison-Wesley, 2010.
4. **Karel Skoupý**, *LaTeX Cookbook*, Packt Publishing, 2015.

Course Code: MCA 536

Name of Course: Java Programming

Course Coordinator: Ajay Kumar

Credit: 4

Course Objectives:

- To develop a deep understanding of OOP concepts, including encapsulation, inheritance, polymorphism, and abstraction, and apply them effectively in Java programming.
- To gain expertise in using essential Java APIs and libraries, such as collections, I/O streams, multithreading enabling the creation of robust and scalable applications.
- To equip students with the skills to develop platform-independent applications using Java's "write once, run anywhere" capability and to understand the architecture of enterprise-level applications.
- To foster the ability to analyze complex problems, design modular and maintainable Java code, and apply best practices for software development.

Course Outcomes: After the successful completion of this course, the student will be able to

- Learn fundamental Java concepts such as object-oriented programming (OOP), classes, objects, inheritance, polymorphism, encapsulation, and abstraction.
- Gain expertise in writing efficient, secure, and scalable Java applications using exception handling, multithreading, and file handling techniques.
- Improve logical thinking and coding proficiency by implementing Java-based solutions in real-world scenarios, including software development and competitive programming.
- Explore Java APIs, collections framework, JDBC for database connectivity, and frameworks like Spring and Hibernate for enterprise application development.

Evaluation Criteria:

A student will be evaluated on the following basis:

- End Semester Examination: 70% weightage
- Continuous Internal Assessment: 30% weightage

Course Content:

Unit-I: Introduction

Module/ Chapter 1: **Introduction to java**

Introduction to java, objectives of java, java features, features of object oriented programming: Abstraction, Encapsulation, Data hiding, Inheritance, and Polymorphism, Concept of Java Virtual Machine

Module/ Chapter 2: **Data types**

Meaning of statically and strongly typed, primitives types: int, float, long, char etc. non-primitive types: class, arrays, and interface etc. introduction to first java program, installation of JDK and VS Code.

Module/ Chapter 3: **Operators**

Basic syntax of java program, Java keywords, Java operators: Arithmetic, Bitwise, Relational, Logical, assignment, conditional and operator precedence table, Type casting

Module/ Chapter 4: **Control Statements**

Selection: if, if-else, Nested if, if-else-if ladder, and switch-case; Iteration: while, do-while, and for; Jump: break, continue, and return

Unit-II: **Java Classes**

Module/ Chapter 5: **Class and object**

Concept of class and object, static member variable, function, and class, Wrapper classes and its methods

Module/ Chapter 6: **Constructor**

Cocept of Packages, meaning of import keyword, Access modifiers: private, public, protected, and default, parameterized methods, constructors, meaning of this keyword, examples of method overloading.

Unit-III: **Inheritance**

Module/ Chapter 7: **Introduction to Inheritance**

Concept of inheritance, extends, super/parent, and sub/child class, Types of inheritance: Single, multilevel, and hierarchical and related examples

Module/ Chapter 8: **Polymorphism**

Polymorphism, Concept of method overriding, meaning of super keyword, difference between method overloading and overriding

Unit-IV: Exception handling

Module/ Chapter 09: **Introduction to Exception**

Concept of exceptions in java, and handling options. Class hierarchy of object class, Types of exceptions, example of try, catch, and finally

Module/ Chapter 10: **Abstract class**

Concept of abstract classes and interfaces, difference between class and interface, Multithreading and thread class.

Unit-V: Advanced Topics

Module/Chapter 11: **Java Packages**

Concept of packages, Defining a packages, access protection, Importing packages

Module/Chapter 12: **Java Interfaces**

Concept of interfaces, Defining a interfaces, implementing interfaces

Module/Chapter 13: **Java Collections**

Collection interface in java, Java list interface, ArrayList, Vector class, LinkedList class, stack class in java

Module/Chapter 14: **Java file handling**

Streams in java, file operations: Create a file, read from a file, write to a file, Delete a file

Suggested Readings:

1. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill
2. "Head First Java", Orielly Media Inc. 2nd Edition, 2005.

3. Cay S. Horstmann, Gary Corness, "Core Java 2 Volume 1 - Fundamentals)", 9th Edition, Printice Hall.
4. Java 2: The Complete Reference, Herbert Schildt, Fifth Edition, Tata McGraw Hill

Course Code: **MCA 603**
Name of Course: **Research Paper Publication**
Course Coordinator: **Dr. Pradeep Chouksey**
Credit: **2**

Course Objectives

- Develop the ability to conceptualize, structure, and articulate a comprehensive research proposal.
- Enhance analytical thinking by identifying relevant research problems and proposing viable research questions.
- Strengthen the understanding of existing literature and apply it to justify research objectives.
- Formulate appropriate research methodologies, including data collection and analysis techniques.
- Demonstrate clear academic writing, proper referencing, and persuasive research argumentation.

Course Outcomes: After successful completion of this course, learners will be able to:

CO1: Develop a clear and coherent understanding of research processes.
CO2: Conduct a comprehensive literature review to justify the research problem.
CO3: Design a robust methodology appropriate to the proposed research objectives.
CO4: Write and present a well-structured academic research proposal adhering to academic standards.
CO5: Defend and critically discuss their research proposal during the viva-voce.

2. Guidelines for Research Paper Publication (RPP)

- The work carried out by the candidate in MCA 604 (Literature Review) and MCA 605 (Research Proposal) should result in a research paper.
- The paper needs to be presented in a national or international conference or published in conference proceedings, peer-reviewed journals, Scopus-indexed journals, or SCI journals.
- Appropriate acknowledgments should be provided to the university in all publications.
- Candidates are encouraged to collaborate with their supervisors for successful publication.
- Compliance with ethical publication standards and plagiarism policies is mandatory.

3. Research Paper Publication Evaluation

Publication Type	Weightage
National Level Conference Presentation	50% - 55%

International Level Conference Presentation	56% - 60%
National Conference Proceedings	61% - 65%
International Conference Proceedings	66% - 70%
Peer-reviewed Journals	71% - 80%
Scopus Indexed Journals	81% - 90%
SCI Journals	91% - 100%

4. Evaluation Breakdown

Mode of Evaluation	Weightage	Marks	Component
Internal Evaluation	30%	30	Continuous Internal Assessment*
External Evaluation	70%	70	• Research Paper Evaluation • Viva Voce
Total	100%	100	-

**The allotted mentors will conducting the internal evaluation (30%) under Continuous Internal Assessment.*

5. Plagiarism Policy

As per UGC (Promotion of Academic Integrity and Prevention of Plagiarism in Higher Educational Institutions) Regulations, 2018, plagiarism is strictly prohibited.

Plagiarism Levels & Penalties

Similarity Percentage	Consequences
Up to 10%	No penalty
10% – 40%	Revision required within a stipulated time
40% – 60%	Resubmission after a 6-month cooling period
Above 60%	Cancellation of dissertation and disciplinary action

6. Final Assessment

The overall score will be based on the successful completion of both the dissertation and viva-voce while ensuring compliance with UGC plagiarism norms.

7. Suggested Readings

- Kumar, R., *Research Methodology: A Step-by-Step Guide for Beginners*, SAGE Publications, 5th Edition, 2021.
- Kothari, C. R., *Research Methodology: Methods and Techniques*, New Age International Publishers, 4th Edition, 2022.
- Creswell, J. W., & Creswell, J. D., *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*, SAGE Publications, 5th Edition, 2018.
- Machi, L. A., & McEvoy, B. T., *The Literature Review: Six Steps to Success*, Corwin Press, 4th Edition, 2022.

- Zobel, J., *Writing for Computer Science*, Springer, 3rd Edition, 2014.
- Dawson, C. W., *Projects in Computing and Information Systems: A Student's Guide*, Pearson, 2nd Edition, 2015.
- Kitchenham, B., & Charters, S., *Guidelines for Performing Systematic Literature Reviews in Software Engineering*, EBSE Technical Report, 2007.
- IEEE, *IEEE Citation Guide*, IEEE, 2021.
- American Psychological Association, *Publication Manual of the American Psychological Association*, APA, 7th Edition, 2020.

Course Code: **MCA 606**
Name of Course: **Project/Dissertation**
Course Coordinator: **Dr. Parveen Sadotra**
Credit: **8**

Course Objectives

- Develop the ability to apply theoretical knowledge to solve real-world problems through a comprehensive project or dissertation.
- Enhance technical, analytical, and problem-solving skills by designing and implementing innovative software solutions.
- Cultivate independent research capabilities, including data analysis, coding proficiency, and system design.
- Strengthen project management skills through effective planning, scheduling, and execution of tasks.
- Promote effective communication through the documentation and presentation of project findings.

Course Outcomes: After successful completion of this course, learners will be able to:

CO1: Identify and analyze complex real-world problems to propose effective software-based or research-oriented solutions.
CO2: Design and develop functional applications or conduct comprehensive research adhering to academic and industry standards.
CO3: Implement software systems or research models using appropriate programming languages, frameworks, and tools.
CO4: Evaluate and test the project solution for functionality, reliability, and security.
CO5: Document project processes and findings systematically and defend their work through effective oral and written communication.

1. Evaluation Breakdown

Component	Credits	Marks	Mode of Evaluation
Project	04	200	External Examiner Evaluation
Viva-Voce	04	200	Online Oral Defense by External Examiner
Total	08	400	-

2. Project Guidelines

2.1 Topic Selection & Approval

- Students must select a topic in consultation with the course coordinator
- Approval must be obtained via email/official online submission
- Topics should align with current industry trends, academic relevance, or societal impact

2.2 Project/Dissertation Execution

- Work must be completed independently (no financial support from the department); Students can seek external funding.
- Students may seek guidance from concerned course cordinator but must ensure originality.
- Timeline: Strict adherence to department deadlines (late submissions may incur penalties)

2.3 Project/Dissertation Documentation Requirements

The project report (100-125 pages, excluding code) must demonstrate original work with specific focus on your implementation. Avoid generic theoretical content from reference books.

Structure of Project Report

Preliminary Sections

- Title page
- Certificate of Originality
- Table of contents with page numbers
- List of figures/tables

Main Body

1. Introduction
    - Problem statement
    - Objectives
    - Scope and limitations
2. System Analysis
    - Requirements specification
    - Feasibility study
    - Project planning (WBS, PERT, Gantt charts etc.)
3. Design Documentation
    - DFD/ER/UML diagrams
    - Database schema (if any)
4. Implementation
    - Module specifications
    - Code organization
    - User interface design
5. Testing
    - Test cases and results
    - Debugging process
    - Security implementation
6. Ancillary Sections
    - Future enhancements
    - Bibliography

- Appendices

Technical Requirements:

1. Code Submission:
    - Complete source code in .zip format
    - README file with setup instructions
2. Testing Evidence: Screenshots of test executions

2.4 Formatting Standards of Project Reports

- Font: Times New Roman
    - Body: 12pt
    - Headings: 14pt
    - Sub Headings: 13pt
    - Title: 16pt
- Spacing: 1.5 line
- Margins: 1 inch
- Page Numbers: Bottom center

2.5 Evaluation Parameters

Criteria	Weightage
Documentation Completeness	25%
Technical Depth	25%
Originality & Innovation	20%
Implementation Quality	20%
Compliance with Guidelines	10%

3. Viva-Voce Guidelines

3.1 Examination Process

- Duration: 30 minutes (15 min presentation + 15 min Q&A)
- Platform: As Decided by CDOE, CUHP

3.2 Evaluation Criteria

Criteria	Weightage
Technical Understanding	30%
Problem-Solving Ability	25%
Presentation Skills	20%
Documentation Defense	15%
Future Scope Analysis	10%

4. Plagiarism Policy

As per UGC (Promotion of Academic Integrity and Prevention of Plagiarism in Higher Educational Institutions) Regulations, 2018, plagiarism is strictly prohibited.

Plagiarism Levels & Penalties

Similarity Percentage	Consequences
Up to 10%	No penalty
10% – 40%	Revision required within a stipulated time
40% – 60%	Resubmission after a 6-month cooling period
Above 60%	Cancellation of dissertation and disciplinary action

5. Final Assessment

The overall score will be based on the successful completion of both the dissertation and viva-voce while ensuring compliance with UGC plagiarism norms.