Scheme of Instruction and Syllabus

For

B.Sc - Computer Science

Under Choice Based Credit System

(With effect from the academic year 2025-26)

Mahatma Gandhi University

Nalgonda
Telangana State
508254

Mahatma Gandhi University

Faculty of Science

B.Sc (COMPUTER SCIENCE)

With effect from the Academic Year 2025-26

Year/ Semester	Type of Course	Paper Title	Teaching Hours/ week		No. of Credits	Max. Marks		
			Theory	Practical		Internal Marks	External marks	Total marks
I/I	DSC-1	Programming in C	4		4	20	80	100
	DSC-1	Programming in C Lab		2	1	--	25	25
I/II	DSC-2	Data Structures using C	4		4	20	80	100
	DSC-2	Data Structures using C Lab		2	1	--	25	25
II/I	DSC-3	Data Base Management System	4		4	20	80	100
	DSC-3	Data Base Management System Lab		2	1	--	25	25
II/II	DSC-4	Programming in Java	4		4	20	80	100
	DSC-4	Programming in Java Lab		2	1	--	25	25

CH. Sudh.

CHAIRPERSON
Board of Studies in
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – I

Course type	Paper Title	Hours per week	Marks	
DSC-1	Programming in C	Theory: 04	Internal	External
		Credit: 04	20	80

Course Objectives:

Cob1: To introduce the basics of programming languages, focusing on the C programming language, and explore different ways of writing and designing algorithms.
Cob2: To understand the key concepts such as variables, data types, operators, control structures, arrays and strings.
Cob3: To emphasize problem-solving techniques using functions and pointers.
Cob4: To learn the concept of user-defined data types and files.

Course Outcomes:

CO1: Understand program structure and design algorithm.
CO2: Develop basic programs by applying concepts such as control structures, arrays and strings.
CO3: Implement functions, pointers and Dynamic Memory Allocation (DMA).
CO4: Create user-defined data types and implement file operations.

Unit I
Programming Fundamentals: Algorithms and Flow charts, Generation and classification of programming languages, Processes involved in program execution: compilation, interpretation, loading and linking.
Basics of C Programming: Introduction to C programming language, Structure of a C program, C tokens, data types, variables, constants, operators, expression evaluation (precedence, associativity), type conversions in C.

Unit II
Input and Output: Non-formatted and formatted input/output functions, Escape sequences and their usage in I/O. Control Statements - Sequence statements, Selection statements: if, if-else, nested if, switch, conditional operators, Iterative statements: while, do-while and for. Special control statements: goto, break, continue, return, and exit.
Arrays and Strings: One-dimensional arrays, Multidimensional arrays and character arrays.

Unit III
Functions: Function definition, declaration and calling mechanisms, types of functions, ctype functions and string functions, call-by-value, call-by-reference. Passing arrays to functions, recursion, inline functions. Scope and lifetime of variables, storage classes.
Pointers: Introduction, address-of operator (&). Uses of pointers, Pointer types: pointers and arrays, pointers and strings, pointer to pointer, array of pointers. Dynamic memory allocation, malloc, calloc and free.

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

Unit IV

User-Defined Data Types: Structures and unions: Definition, initialization, accessing members, arrays of structures, structures vs. unions, enumeration types (enum).

File handling: Introduction, file operations, file functions: open, close, read and write. Working with text and binary files.

Suggested Books

1. Reema Thareja, 'Programming in C', Oxford University Press, Second Edition, 2016.
2. Kernighan, B.W and Ritchie, D.M, 'The C Programming language', Second Edition, Pearson Education, 2015.

Reference Books:

1. Ivor Horton, Beginning C
2. Ashok Kamthane, Programming in C
3. Herbert Schildt, The Complete Reference C
4. Paul Deitel, Harvey Deitel, C How to Program
5. R.S.Bichkar, "Programming with C" University Press, 2024.
6. Byron S. Gottfried, Theory and Problems of Programming with C
7. Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language
8. B. A. Forouzan, R. F. Gilberg, A Structured Programming Approach Using C

CHAIRPERSON
Board of Studies in
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – I

Course type	Paper Title	Hours per week	Marks	
DSC-1	Programming in C Lab	Practical: 02	Internal	External
		Credits: 01	--	25

Recommended Software: GCC on Linux, DevC++ or Code Blocks on Windows 10.
External Examination Requirements: Students need to demonstrate the execution of two programs in the external lab exam.

Course Objectives:

Cobl: Develop fundamental programming skills in C by implementing conditional statements, loops, functions, and data structures for problem-solving.
Cob2: Apply key C programming concepts such as arrays, pointers, strings, file handling, and recursion to build efficient algorithms for real-world applications.

Course Outcomes:

COl: Demonstrate the ability to write, debug, and execute C programs for solving mathematical and logical problems using control structures, functions, and data structures.
CO2: Apply fundamental C programming concepts, including file handling, recursion, and memory management, to develop efficient solutions for computational tasks.

Lab Experiments:

1. Write a C program to input numbers and find the largest of two or three numbers using if statements and the conditional (ternary) operator (? :). Display the largest number.
2. Write a C program that takes an integer input and outputs the reversed number.
3. Write a C program to print all prime numbers between 2 and a given number n.
4. Write a C program to find the roots of a quadratic equation $ax^2+bx+c=0$.
5. Write a C program to print a triangle pattern of stars (*), where the number of lines is given by the user.
6. Write a C program to find the largest and smallest elements in an array of n numbers.
7. Write a C program to multiply two matrices of 3x3.
8. Write a C program to find the Greatest Common Divisor (GCD) of two numbers using both iteration and recursion.
9. Write a C program to demonstrate the use of different storage classes (auto, register, static,

CHAIRPERSON
Board of Studies in
Computer Science
Mahatma Gandhi University, NLG-508 254.

extern).

10. Write a C program to demonstrate the concepts of call-by-value and call-by-reference.
11. Write a C program that takes a string from the command-line arguments and counts the occurrence of each alphabet letter in the string.
12. Write a C program to demonstrate the usage of the enum data type.
13. Write a C program that demonstrates various string functions from the <string.h> library.
14. Write a C program that demonstrates structures and unions.
15. Write a C program that opens a file and counts the total number of characters in it.
16. Write a C program that copies content from an existing text file to a new file.

CH.

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – II

Course type	Paper Title	Hours per week	Marks	
DSC-2	Data Structures Using C	Theory: 04	Internal	External
		Credits: 04	20	80

Course Objectives:

Cob1: To discuss the linear data structures and their applications.
Cob2: To Understand Queues, Linked list and Hashing Concepts.
Cob3: To understand and implement trees and graphs with efficient traversal, searching, and optimization techniques.
Cob4: Analyze and implement advanced searching and sorting techniques, including hashing and overflow handling, to optimize data organization and retrieval.

Course Outcomes:

CO1: Understand and implement fundamental data structures, including arrays and stacks, for efficient data manipulation and expression evaluation.
CO2: Apply linked lists, queues, and hashing techniques to optimize data storage, retrieval, and processing.
CO3: Analyze and implement tree and graph structures, including traversal techniques and efficient searching strategies.
CO4: Develop and optimize searching and sorting algorithms to enhance data organization and retrieval efficiency.

UNITI

Introduction to Data structures: Definition, Types of Data structures. **Arrays:** Arrays - ADT, ordered lists, sparse matrices, representation of arrays. **Stacks:** Stack ADT, Stacks using Arrays, Stacks using dynamic arrays, Evaluation of Expressions - Evaluating Postfix Expression, Infix to postfix expression, checking well-formed parenthesis, reversing a string.

UNIT II

Queues: Queues ADT, operations, Circular Queues, Applications. **Linked Lists:** Singly Linked Lists and Chains, Linked Stacks and Queues, Polynomials, Operations for circularly linked lists, Equivalence Classes, Doubly Linked Lists. **Hashing:** Static Hashing, Hash Tables, Hash Functions, Overflow Handling, Theoretical Evaluation of Overflow Techniques.

UNIT III

Trees: Introduction, Binary Trees, Binary Tree Traversals, Heaps, Binary Search trees (BST): Definition, Searching an element, Insertion into a BST, and Deletion from a BST, Efficient Binary Search Trees. AVL Trees: Definition, Insert, search and delete operations.
Graphs: Graph Abstract Data Type, Elementary Graph operations, Graph Traversal Techniques -

CH. Sudha

CHAIRPERSON
Board of Studies in
Computer Science
Mahatma Gandhi University, NLG-508 254.

DFS and BFS, Minimum Cost Spanning Trees - Prim's and Kruskal's Algorithms.

UNIT IV

Searching and Sorting: Sequential search, Binary search, Hash Tables: Hashing Functions, Overflow Handling, Theoretical Evaluation of Overflow Techniques. Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Heap sort, List and Table Sorts.

Suggested Books:

1. Horowitz E, Sahni S and Susan Anderson-Freed, Fundamentals of Data structures in C, 2nd Edition (2008), Universities Press.

Reference Books:

1. Mark A Weiss, Data Structures and Algorithm Analysis In C, Second Edition (2002), Pearson
2. Kushwaha D. S and Misra A.K, Data structures A Programming Approach with C, Second Edition (2014), PHI.
3. ilberg R. F and Forouzan B. A, Data structures: A Pseudocode Approach with C, Second Edition (2007), Cengage Learning
4. Tanenbaum A. M , Langsam Y. Augenstein M. J, Data Structures using C, Second Edition (2008), Pearson.
5. Thomas H. Connen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, Introduction to Algorithms, Third Edition (2009). MIT Press
6. Chandan Banerjee and Atanu Das, "Data Structures and Algorithms in C and PYTHON", University Press, 2023.
7. YedidyahLangsam, Moshe J. Augenstein, Aaron M. Tenenbaum, Data Structures Using C and C++, Second Edition (2009), PHI

Ch. Sudha

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – II

Course type	Paper Title	Hours per week	Marks	
DSC-2	Data Structures Using C Lab	Practical: 02	Internal	External
		Credits: 01	--	25

Course Objectives:

Cob1: Develop and implement various data structures such as arrays, linked lists, stacks, queues, trees, and graphs for efficient data manipulation.
Cob2: Apply sorting, searching, and hashing techniques to solve computational problems effectively.

Course Outcomes:

CO1: Implement fundamental and advanced data structures, including arrays, linked lists, stacks, queues, trees, and graphs.
CO2: Apply efficient searching, sorting, and hashing techniques to solve computational problems.

Lab Experiments (Using C programming Language):

1. Implementation of Stacks and Queues using Arrays.
2. Implementation of Circular Queue.
3. Implementation of Infix to Postfix Conversion, Postfix Expression Evaluation.
4. Implementation of Singly Linked List
5. Implementation of Doubly Linked List.
6. Implementation of Circular Linked List.
7. Implementation of Stacks using Linked Lists
8. Implementation of Queues using Linked Lists.
9. Implementation of Linear search and Binary Search.
10. Implementation of Operations on Binary Tree
11. Implementation of Binary Search Tree.
12. Implementation of Traversal on Graphs.
13. Implementation of Selection, Bubble and Insertion Sort.
14. Implementation of Merge Sort.
15. Implementation of Quick Sort.

CHAIRPERSON
Board of Studies in
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – III

Course type	Paper Title	Hours per week	Marks	
DSC-3	Data Base Management System	Theory: 04	Internal	External
		Credits: 04	20	80

Course Objectives:

Cob1: Understand the fundamental concepts, architecture, and advantages of database management systems over traditional file-based systems.
Cob2: Design efficient database schemas using the Entity-Relationship model and apply normalization techniques to minimize redundancy.
Cob3: Develop SQL queries and PL/SQL programs for efficient data manipulation, retrieval, and transaction control in relational databases.
Cob4: Implement transaction management, concurrency control, and security mechanisms to ensure data integrity and secure database operations.

Course Outcomes:

CO1: Explain fundamental concepts, architecture, advantages of Database Management Systems.
CO2: Design and normalize relational database schemas using the Entity-Relationship model and normalization techniques.
CO3: Develop SQL and PL/SQL queries to perform data manipulation, retrieval and transaction management.
CO4: Implement concurrency control, recovery mechanisms, and security measures to ensure data integrity and protection.

Unit I
Introduction to Database Management System: Data, Information, Metadata, Database, Database Management System, File Based System, Drawbacks of File-Based System, The Database approach, Components of Database Environment, Advantage of DBMS, Three-Tier Architecture, Database Languages, Data Models, Database Users, Database Administrator, Types of Databases.

Unit II
Entity-Relationship Model: Introduction, Components of ER Diagrams, Entities, Attributes, Keys, Strong and Weak entity sets, Relationships, Types of Entities, Types of Attributes, Types of Relationships, Degree of a relationship, Cardinality. Generalization and Specialization, Aggregation and Composition, Transforming E-R model to Relational model.
Normalization: Data Redundancy, Functional Dependencies, Basic Normal Forms: First Normal Form (INF), Second Normal Form (2NF), Third Normal Form (3NF), Boyce Codd Normal Form (BCNF), De-normalization.

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

Unit III

Structured Query Language: Basic form of SQL query, Data Types, Integrity Constraints, Null Values, Commit, rollback, Logical operators, Special Operators, Order by, Group by and Having clause, Aggregate Functions, Set Operators, Alias, Joins, Nested Sub queries, Correlated Sub queries, Views: Create a view, Delete a View.

PL/SQL: Introduction, Structure of PL/SQL, Elements, Data Types, Control Structures, Steps to Create a PL/SQL Program, Iterative Control Cursors, Steps to Create a Cursor, Procedures, Functions, Packages, Exception Handling, Database Triggers, Types of triggers.

Unit IV

Transaction Management: What is transaction, Properties of Transactions, Database, Concurrency control, serializability, recoverability, Concurrency control with locking methods, concurrency control with time stamping methods, Concurrency control with optimistic methods, Deadlock. Database Recovery: The Need for Recovery, Transactions and Recovery, Recovery Facilities, Recovery Techniques. Database Security: Threats, Computer Based Controls: Authorization, Access Controls, Views, Backup and Recovery, Integrity, Encryption, RAID.

Suggested Books:

1. Abraham Silberschatz, H. Korth and S. Sudarshan, Database System Concepts, 6th Ed., Tata McGraw Hill, 2011
2. Thomas M. Connolly, Carolyn E. Begg, Database Systems-A Practical Approach to Design, Implementation, and Management (6e)
3. Ivan Bayross, SQL, PL/SQL: The Programming Language of Oracle, 4th Revised Edition with CD-ROM by Ivan Bayross, BPB Publications.

Ravi

Jayde

CH. Sudhe

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – III

Course type	Paper Title	Hours per week	Marks	
DSC-3	Data Base Management System Lab	Theory: 02	Internal	External
		Credits: 01	--	25

Course Objectives:

Cob1: Develop and manage relational database schema by creating tables, defining primary and foreign keys, and performing data manipulation using SQL queries.

Cob2: Implement PL/SQL programming concepts such as procedures, functions, triggers, and exception handling for efficient database management.

Course Outcomes:

CO1: Construct and execute complex SQL queries to retrieve, modify, and analyze data based on real-world scenarios,

CO2: Apply PL/SQL programming techniques to automate database operations and enforce business rules.

Lab Experiments:

1. Create a database having three tables to store the details of students of Computer Department in your college:

   Student (Roll number, Name of student, Date of birth, Address, Marks, Phone number)

   Paper (Paper code, Name of the Paper)

   Attendance (College roll number, Paper Code, Attendance, Marks).

   a) Identify primary and foreign keys. Create the tables and insert at least 5 records in each table.

   b) Design a query that will return the records from the table name Paper along with the name of student from the first table, related to students who have more than 75% attendance and more than 60% marks in paper2.

   c) List all students who live in "Warangal" and have marks greater than 60 in paper 1.

   d) Find the total attendance and total marks obtained by each student.

   e) List the names of all the students who secured highest marks in paper2.

2. Create the following tables, enter at least 5 records in each table and answer the queries given below:

   Employee (Person_Name, Street, City)

CH. Sudha

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

Works (Person_Name, Company_Name, Salary)

Company (Company_Name, City)

Manages (Person_Name, Manager_Name)

a) Identify primary and foreign keys.

b) Alter table employee, add a column "email" of type varchar(20).

c) Find the name of all managers who work for both Samba Bank and NCB Bank.

d) Find the names, street address and cities of residence and salary of all employees who work for "Samba Bank" and earn more than $10,000.

e) Find the names of all employees who live in the same city as the company for which they work.

f) Find the highest salary, lowest salary and average salary paid by each company.

g) Find the sum of salary and number of employees in each company.

h) Fin<l the name of the company that pays highest salary.

3. Create the following tables, enter at least 5 records in each table and answer the queries given below.

Suppliers (SNo, Sname, Status, SCity)

Parts (PNo, Pname, Colour, Weight, City) Project (JNo, Jname,Jcity)

Shipment (Sno, Pno, Jno, Quantity)

a) Identify primary and foreign keys.

b) Get supplier numbers for suppliers in Paris with status>20.

c) Get suppliers details for suppliers who supply part P2. Display the supplier list in increasing order of supplier numbers.

d) Get suppliers names for suppliers who do not supply part P2.

e) For each shipment get full shipment details, including total shipment weights.

f) Get all the shipments where the quantity is in the range 300 to 750 inclusive.

g) Get part nos. for parts that either weigh more than 16 pounds or are supplied by suppliers S2, or both.

h) Get the names of cities that store more than five red parts.

i) Get full details of parts supplied by a supplier in Hyderabad.

j) Get part numbers for part supplied by a supplier in Warangal to a project in Chennai.

k) Get the total number of project supplied by a supplier (say, S1).

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

l) Get the total quantity of a part (say, PI) supplied by a supplier (say, SI).

4. Write a PL/SQL Program to demonstrate Procedure.
5. Write a PL/SQL Program to demonstrate Function.
6. Write a PL/SQL program to Handle Exceptions.
7. Write a PL/SQL Program to perform a set of DML Operations.
8. Create a View using PL/SQL program.
9. Write a PL/SQL Program on Statement Level Trigger.
10. Write a PL/SQL Program on Row Level Trigger.

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – IV

Course type	Paper Title	Hours per week	Marks	
DSC-4	Programming in Java	Theory: 04	Internal	External
		Credits: 04	20	80

Course Objectives:

Cob1: To demonstrate an understanding of the fundamental principles of object-oriented programming in Java by defining classes, calling methods, and utilizing class libraries.

Cob2: To analyze and apply concepts of interfaces, packages, exception handling, and multithreading in Java.

Cob3: To apply the Collection Framework and event handling to solve real-world problems in Java.

Course Outcomes:

CO1: Design and Implement Java programming fundamentals, including syntax, object-oriented concepts, and string handling.

CO2: Apply inheritance, polymorphism, interfaces, and exception handling to develop modular and reusable Java applications.

C03. Implement multithreading, collections framework, and file handling for efficient Java programming.

CO4: Develop Java applications that incorporate Swing, JavaFX for creating visually appealing user interfaces

UNIT1

Introduction to Java: Features of Java, bytecode, Structure of java program, language Fundamentals - tokens (comments, Identifiers, Keywords), data types, variables and types of variables (reference, local, static, final), command line arguments, operators, expressions, type conversion and casting, control statements.

Object-Oriented Programming (OOP) Basics: Classes and Objects - Concepts of classes, objects, constructors, methods (types of methods), access control (public, private, protected and default). this keyword, garbage collection, nested and inner classes, String, StringBuffer and StringBuilder.

UNIT II

Inheritance: Inheritance concept, types of inheritance, Member access rules, use of super and final. Abstract classes and methods, **Polymorphism** - method overriding (Runtime-time Polymorphism) and method overloading (Compile-time Polymorphism), overloading constructors **Interfaces:** Defining an interface, implementing interfaces, extending interface. **Packages:** inbuilt packages, user defined packages - Defining, Creating and Accessing a Package, importing packages.

Exception handling: Benefits of exception handling, Exception hierarchy, exception Vs error, classification, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.

and finally, re throwing exceptions, built in exceptions, implementing custom exceptions.

UNIT III

Multithreading: Java Thread Model - Thread, Thread Life Cycle, The Main Thread, Ways to create a Thread - Implementing Runnable interface, Extending Thread class, creating multiple threads, thread priorities, synchronization.

Collections: Overview of Java Collection frame work, **java.util** - Collection classes - Array List, Linked List, Hash Set, Tree Set, Collection Interfaces - Collection, List, Set, and Accessing Collection via iterator, working with Map. Other Utility classes: String Tokenizer, Date, Calendar, Gregorian calendar, Scanner.

Java Input/Output: exploring java.io, Java 1/0 classes and interfaces, File, Stream classes, byte stream, character stream, serialization.

UNIT IV

Swing: JFrame, JLabel, ImageIcon, JTextField, JPasswordField, the Swing buttons, JTabbedpane, JScroll-Pane, JList, JComboBox, Event Handling: Event. Classes, Source of Events, Event Listener Interfaces. Handling mouse and keyboard events.

JavaFX: Introduction, Setup, Scene Graph (root, children, hierarchy). Layout containers -Vbox, HBox, GridPane, StackPane, Buttons, controls, Styling in JavaFX, Animation and Transitions in JavaFX.

Suggested Books:

1. Herbert Scheidt, "The Complete Reference Java, 11th Edition, Tata McGraw Hill, 2021.
2. James M Slack, Programming and Problem Solving with JAVA, Thomson Learning, 2002.
3. C Thomas Wu, An Introduction to Object Oriented Programming with Java 6th Edition, McGraw Hill Publishing, 2019.
4. H. M. Dietel and P. J. Dietel, Java How to Program, Sixth Edition, Pearson Education/PHI
5. Carl Dea, Mark Heckler, Gerrit Grunwald, Jose Pereda, " JavaFX 8: Introduction by Example", 1st Edition, Apress - 2014

CH. Sudh

CHAIRPERSON
Board of Studies in
Computer Science
Mahatma Gandhi University, NLG-508 254.

B. Sc. Computer Science

SEMESTER – IV

Course type	Paper Title	Hours per week	Marks	
DSC-4	Programming in Java Lab	Practical: 02	Internal	External
		Credits: 01	--	25

Course Objectives:

Cob1: Develop a strong foundation in Java programming, covering core concepts such as object-oriented principles, exception handling, multithreading, and collections.
Cob2: Design and implement interactive Java applications using GUI frameworks like Swing and JavaFX, incorporating event handling and advanced Java functionalities.

Course Outcomes:

CO1: Apply object-oriented programming concepts, exception handling, and multithreading to develop efficient and scalable Java applications.
CO2: Design interactive user interfaces using Swing and JavaFX while implementing event handling and advanced Java utilities

Lab Experiments:

1. Write a Java program that demonstrates the concept of a class, along with method overloading.
2. Write a Java program that reads a line of integers, displays each integer, and calculates the sum of all the integers.
3. Write a Java program to demonstrate single-level and multi-level inheritance.
4. Write a Java program to showcase the use of interfaces and abstract classes.
5. Write a Java program to implement exception handling.
6. Write a Java program that illustrates the concept of threads using Thread class and runnable interface.
7. Write a Java program to demonstrate thread synchronization.
8. Write a java program to create a package and import the classes from the defined package.
9. Write a Java program to illustrate the use of collection classes, such as ArrayList and LinkedList.
10. Write a Java program that demonstrates the concept of I/O streams.
11. Write a Java program to implement the concept of serialization.
12. Write a Java program that functions as a simple calculator. Use a grid layout to arrange buttons for digits and operations (+,-,*, %), and include a text field to display the result.

CHAIRPERSON
Board of Studies in
Computer Science
Mahatma Gandhi University, NLG-508 254.

13. Create a simple user profile form using Swing, with fields for name, age, and a profile picture.

14. Develop a JavaFX application that demonstrates the following core concepts:

    a.Use a VBox as the root node to establish a basic scene graph hierarchy.

    b.Utilize layout containers such as HBox, GridPane, and StackPane to organize UI elements.

    c.Incorporate Button, Label, and TextField controls for user interaction

CH. Sudha

CHAIRPERSON
Board of Studies In
Computer Science
Mahatma Gandhi University, NLG-508 254.