

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**B.Sc. – I Semester**

**Course: PROBLEM SOLVING IN C**

**Course Code:**  
**Paper : I**

**No. of Hours/Week: 4**

**Course Objective:**

To explore the Fundamentals of computers and reinforce computer vocabulary, particularly with respect to personal use of computer hardware and software. To impart basic knowledge of C Programming language so that Students will be able to develop logics and applications to solve real time problems using C.

**Course Outcomes:**

At the end of the course the student will be able to

1. Understand the fundamentals of C programming.
2. Make use of loops, decision making statements and functions to solve the problem.
3. Implement different Operations on Arrays.
4. Understand Pointers, Structures and Unions.
5. Implement File Operations for a given application using C file handling functions.

**UNIT I**

**General Fundamentals:** Introduction to computers: Block diagram of a computer, characteristics and limitations of computers, applications of computers, types of computers, computer generations.

**Introduction to Algorithms and Programming Languages:** Algorithm – Key features of Algorithms, Flow Charts, Programming Languages – Generations of Programming Languages – Structured Programming Language- Design and Implementation of Correct, Efficient and Maintainable Programs.

**UNIT II**

**Introduction to C:** Introduction – Structure of C Program – Writing the first C Program –File used in C Program – Compiling and Executing C Programs – Using Comments – Keywords – Identifiers – Basic Data Types in C – Variables – Constants – I/O Statements in C- Operators in C- Programming Examples.

**Decision Control and Looping Statements:** Introduction to Decision Control Statements– Conditional Branching Statements – Iterative Statements – Nested Loops – Break and Continue Statement – Goto Statement

**UNIT III**

**Arrays:** Introduction – Declaration of Arrays – Accessing elements of the Array – Storing Values in Array– Operations on Arrays – one dimensional, two dimensional and multi dimensional arrays, character handling and strings.

## **UNIT IV**

**Functions:** Introduction – using functions – Function declaration/ prototype – Function definition – function call – return statement – Passing parameters – Scope of variables – Storage Classes – Recursive functions.

**Structure, Union, and Enumerated Data Types:** Introduction – Nested Structures – Arrays of Structures – Structures and Functions– Union – Arrays of Unions Variables – Unions inside Structures – Enumerated Data Types.

## **UNIT V**

**Pointers:** Understanding Computer Memory – Introduction to Pointers – declaring Pointer Variables – Pointer Expressions and Pointer Arithmetic – Null Pointers - Passing Arguments to Functions using Pointer – Pointer and Arrays – Memory Allocation in C Programs – Memory Usage – Dynamic Memory Allocation – Drawbacks of Pointers

**Files:** Introduction to Files – Using Files in C – Reading Data from Files – Writing Data to Files – Detecting the End-of-file – Error Handling during File Operations – Accepting Command Line Arguments.

### **Additional Inputs:**

C Pre-processor, Conditional Compilation, Header Files, Sparse Matrices, Arrays of Union Variables, Unions inside Structures.

### **Text Books:**

1. E Balagurusamy – Programming in ANSIC – Tata McGraw-Hill publications.
2. Brain W Kernighan and Dennis M Ritchie - The ‘C’ Programming language” - Pearson publications.
3. Ashok N Kamthane: Programming with ANSI and Turbo C, Pearson Edition Publications.
4. YashavantKanetkar - Let Us ‘C’ – BPB Publications.

### **Reference Books:**

1. Introduction to C programming by REEMA THAREJA, OXFORD UNIVERSITY PRESS.
2. Schaum's Outline of Programming with C, by Byron Gottfried, 2nd Edition, (Indian Adapted Edition), TMH publications, New Delhi, 2006.

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**B.Sc. – I Semester**

**Course: PROBLEM SOLVING IN C**

**Course Code:**

**No. of Hours/Week: 2**

**Course Objective:**

To develop programming skills using the fundamentals of C Language and to enable effective usage of arrays, structures, functions, pointers and to implement the memory management concepts.

**Course Outcomes:**

At the end of the course the student will be able to

1. Implement programs using fundamental features of C Language.
2. Solve problems with the use of loops, decision making statements and functions.
3. Implement programs performing various Operations on Arrays.

**List of Experiments**

1. Write a program to check whether the given number is Armstrong or not.
2. Write a program to find the sum of individual digits of a positive integer.
3. Write a program to generate the first n terms of the Fibonacci sequence.
4. Write a program to find both the largest and smallest number in a list of integer values
5. Write a program to demonstrate refaction of parameters in swapping of two integer values using **Call by Value&Call by Address**
6. Write a program that uses functions to add two matrices.
7. Write a program to calculate factorial of given integer value using recursive functions
8. Write a program for multiplication of twoN X N matrices.
9. Write a program to perform various string operations.
10. Write a program to search an element in a given list of values.
11. Write a program to sort a given list of integers in ascending order.
12. Write a program to calculate the salaries of all employees using *Employee (ID, Name, Designation, Basic Pay, DA, HRA, Gross Salary, Deduction, Net Salary)* structure.
  - a. DA is 30 % of Basic Pay
  - b. HRA is 15% of Basic Pay
  - c. Deduction is 10% of (Basic Pay + DA)
  - d. Gross Salary = Basic Pay + DA+ HRA
  - e. Net Salary = Gross Salary - Deduction

13. Write a program to illustrate pointer arithmetic.
14. Write a program to read the data character by character from a file.
15. Write a program to create **Book (ISBN, Title, Author, Price, Pages, Publisher)** structure and store book details in a file and perform the following operations
  - a. Add book details
  - b. Search a book details for a given ISBN and display book details, if available
  - c. Update a book details using ISBN
  - d. Delete book details for a given ISBN and display list of remaining Books

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**B.Sc. – II Semester**

**Course: DATA STRUCTURES USING C**

**Course Code:**

**No. of Hours/Week: 4**

**Paper : II**

**Course Objective:**

To introduce the basic concepts of Data Structures and inculcate Knowledge on various types of Data Structures. Also to provide exposure on various Searching and Sorting Techniques.

**Course Outcomes:**

At the end of the course the student will be able to

1. Understand fundamental concepts of Data structures and to design Linked lists.
2. Implement linear data structures stacks, queues.
3. Design non-linear data structures like trees, graphs and implement their operations.
4. Compare and Contrast different searching and sorting techniques.
5. Have knowledge on Data Structures basic operations like insert, delete, search, update and traversal
6. Design and develop programs using various data structures

**UNIT I**

**Introduction to Data Structures:** Introduction to the Theory of Data Structures, Data Representation, Abstract Data Types, Data Types, Primitive Data Types, Data Structure and Structured Type, Atomic Type, Difference between Abstract Data Types, Data Types, and Data Structures, Refinement Stages

**Principles of Programming and Analysis of Algorithms:** Software Engineering, Program Design, Algorithms, Different Approaches to Designing an Algorithm, Complexity, Big 'O' Notation, Algorithm Analysis, Structured Approach to Programming, Recursion, Tips and Techniques for Writing Programs in 'C'

**UNIT II**

**Arrays:** Introduction to Linear and Non- Linear Data Structures, One- Dimensional Arrays, Array Operations, Two- Dimensional arrays, Multidimensional Arrays, Pointers and Arrays, an Overview of Pointers

**Linked Lists:** Introduction to Lists and Linked Lists, Dynamic Memory Allocation, Basic Linked List Operations, Doubly Linked List, Circular Linked List, Atomic Linked List, Linked List in Arrays, Linked List versus Arrays

**UNIT III**

**Stacks:** Introduction to Stacks, Stack as an Abstract Data Type, Representation of Stacks through Arrays, Representation of Stacks through Linked Lists, Applications of Stacks, Stacks and Recursion

**Queues:** Introduction, Queue as an Abstract data Type, Representation of Queues, Circular Queues, Double Ended Queues- Deques, Priority Queues, Application of Queues

#### **UNIT IV**

**Binary Trees:** Introduction to Non- Linear Data Structures, Introduction Binary Trees, Types of Trees, Basic Definition of Binary Trees, Properties of Binary Trees, Representation of Binary Trees, Operations on a Binary Search Tree, Binary Tree Traversal, Counting Number of Binary Trees, Applications of Binary Tree

#### **UNIT V**

**Searching and sorting:** Sorting – An Introduction, Bubble Sort, Insertion Sort, Merge Sort, Searching – An Introduction, Linear or Sequential Search, Binary Search, Indexed Sequential Search

**Graphs:** Introduction to Graphs, Terms Associated with Graphs, Sequential Representation of Graphs, Linked Representation of Graphs, Traversal of Graphs, Spanning Trees, Shortest Path, Application of Graphs.

#### **Additional Inputs:**

Polynomial Representation using Linked List, Drawback of Binary Search Trees, Right Skewed and Left Skewed Binary Search Trees, Balanced Trees: AVL Trees.

#### **Text Books:**

1. “Data Structures using C”, ISRD group Second Edition, TMH
2. “Data Structures through C”, YashavantKanetkar, BPB Publications
3. “Data Structures Using C” Balagurusamy E. TMH

#### **Reference Books:**

1. Data Structures Using C++ by D S Malik, Thomson, India Edition 2006.
2. Data Structures, Algorithms and Applications in C++ by Sahni S, McGraw-Hill, 2002.
3. Classic Data Structures by Samanta.D,Prentice-Hall of India, 2001.
4. Data Structures and Algorithms with Object-Oriented Programming by Heilman G I,Tata McGraw-Hill. 2002. (Chapters I and 14).
5. Introduction to Data Structures with Applications by Tremblay P. and Sorenson P. G.,Tata McGraw-Hill,

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**B.Sc. – II Semester**

**Course: DATA STRUCTURES USING C LAB**

**Course Code:**

**No. of Hours/Week: 2**

**Course Objective:**

To enable the students implement various linear and Non-Linear Data Structures.

**Course Outcomes:**

At the end of the course the student will be able to

1. Implement various operations on arrays
2. Implement Linked list and Perform operations on it.
3. Make use of arrays and linked lists to implement Stack and Queues.
4. Implement various traversals on Trees and Graphs.
5. Implement various shortest path algorithms.
6. Implement various searching and sorting techniques.

**List of Experiments**

1. Write a program to read 'N' numbers of elements into an array and also perform the following operation on an array
  - a. Add an element at the beginning of an array
  - b. Insert an element at given index of array
  - c. Update an element using a value and index
  - d. Delete an existing element
2. Write a program using stacks to convert a given
  - a. postfix expression to prefix
  - b. prefix expression to postfix
  - c. infix expression to postfix
3. Write Programs to implement the Stack operations using an array
4. Write Programs to implement the Stack operations using Linked List.
5. Write Programs to implement the Queue operations using an array.
6. Write Programs to implement the Queue operations using Linked List.
7. Write a program for arithmetic expression evaluation.
8. Write a program for Binary Search Tree Traversals.
9. Write a program to implement dequeue using a doubly linked list.
10. Write a program to search an item in a given list using the following Searching Algorithms
  - a. Linear Search

b. Binary Search.

11. Write a program for implementation of the following Sorting Algorithms

a. Bubble Sort

b. Insertion Sort

c. Quick Sort

12. Write a program for polynomial addition using single linked list

13. Write a program to find out shortest path between given Source Node and Destination Node in a given graph using Dijkstra's algorithm.

14. Write a program to implement Depth First Search graph traversals algorithm

15. Write a program to implement Breadth First Search graph traversals algorithm



**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**II B.Sc. – III Semester**

**Course: DATA BASE MANAGEMENT SYSTEM**

**Course Code :**

**No. of Hours/Week: 4**

**Paper : III**

**Course Objective:**

To present an introduction to database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information from a DBMS.

**Course Outcomes:**

At the end of the course the student will be able to

1. Understand DBMS concepts, data models and Architecture.
2. Understand ER concepts and ER mapping to relational model
3. Improve the database design by normalization.
4. Make use of SQL to retrieve and maintain relational database.
5. Illustrate various constructs in PL/SQL.

**UNIT - I**

**Overview of Database Management System:** Introduction to data, information, database, database management systems, file-based system, Drawbacks of file-Based System, database approach, Classification of Database Management Systems, advantages of database approach, Various Data Models, Components of Database Management System, three schema architecture of data base, costs and risks of database approach.

**UNIT - II**

**Entity-Relationship Model:** Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, **IS A** relationship and attribute inheritance, multiple inheritance, constraints on specialization and generalization, advantages of ER modelling.

**UNIT III**

**Relational Model:** Introduction, CODD Rules, relational data model, concept of key, relational integrity, relational algebra, relational algebra operations, advantages of relational algebra, limitations of relational algebra, relational calculus, tuple relational calculus, domain relational Calculus (DRC), Functional dependencies and normal forms upto 3<sup>rd</sup> normal form.

**UNIT IV**

**Structured Query Language:** Introduction, History of SQL Standard, Commands in SQL, Data Types in SQL, Data Definition Language, Selection Operation, Projection Operation, Aggregate

functions, Data Manipulation Language, Table Modification Commands, Join Operation, Set Operations, View, Sub Query.

## **UNIT V**

**PL/SQL:** Introduction, Shortcomings of SQL, Structure of PL/SQL, PL/SQL Language Elements, Data Types, Operators Precedence, Control Structure, Steps to Create a PL/SQL, Program, Iterative Control, Procedure, Function, Database Triggers, Types of Triggers.

### **Additional Inputs:**

**Transaction Management and Concurrency Control:** What is transaction, Concurrency control, Concurrency control with locking Methods, Concurrency control with time stamping methods.

### **Text Books:**

1. Database System Concepts by Abraham Silberschatz, Henry Korth, and S. Sudarshan, McGrawhill, 2010.
2. Database Management Systems by Raghu Ramakrishnan, McGrawhill, 2002.
3. Fundamentals of Relational Database Management Systems by S. Sumathi, S. Esakkirajan, Springer Publications.
4. SQL: The Ultimate Beginners Guide by Steve Tale.

### **Reference Books:**

1. An Introduction to Database Systems by Bipin C Desai
2. Principles of Database Systems by J. D. Ullman
3. Fundamentals of Database System by R. Elmasri and S. Navathe
4. Database Systems Design, Implementation and Management by Peter Rob, Carlos CoronelSeventh Edition, Thomson , 2007.

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**II B.Sc. – III Semester**

**Course: DATA BASE MANAGEMENT SYSTEMS LAB**

Course Code :

No. of Hours/Week: 2

**Course Objective:**

To provide a strong formal foundation in database concepts and emphasis is on practice to the students to groom them into well-informed database application developers.

**Course Outcomes:**

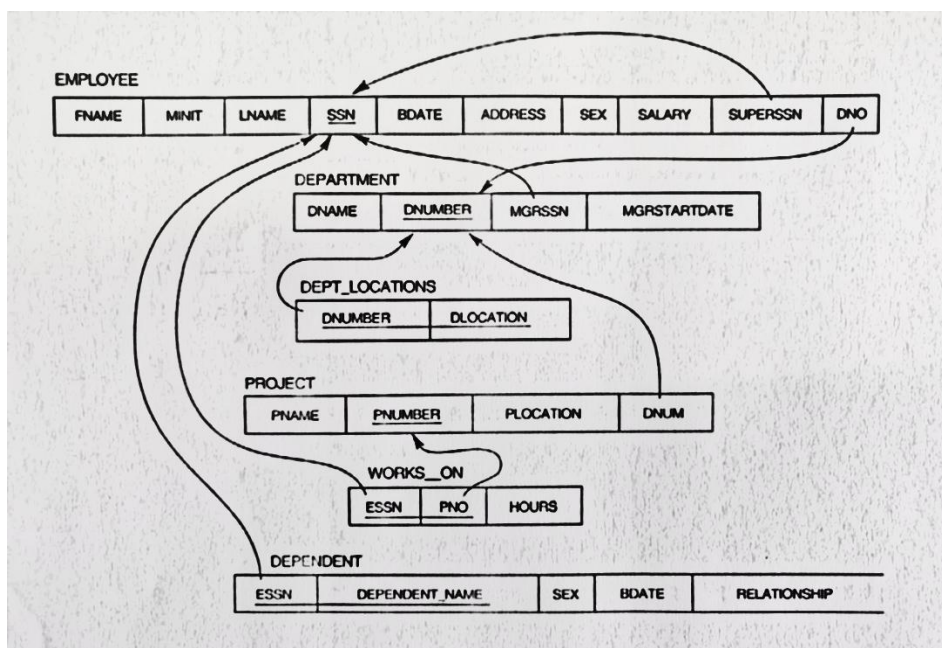
At the end of the course the student will be able to

1. Design database and ER diagrams for the real world scenarios
2. Understand ER concepts and ER mapping to relational model
3. Make use of SQL and PL/SQL to efficiently retrieve and maintain relational database.

**List of Experiments**

1. Draw ER diagram for hospital administration
2. Creation of college database and establish relationships between tables
3. Relational database schema of a company is given in the following figure.

**Relational Database Schema – COMPANY**



## Questions to be performed on the previous schema

1. Create above tables with relevant **Primary Key, Foreign Key and other constraints**
2. Populate the tables with data
3. Display all the details of all employees working in the company.
4. Display **ssn, lname, fname, address** of employees who work in department no 7.
5. Retrieve the **Birthdate and Address** of the employee whose name is 'Franklin T. Wong'
6. Retrieve the name and salary of every employee
7. Retrieve all distinct salary values
8. Retrieve all employee names whose address is in 'Bellaire'
9. Retrieve all employees who were born during the 1950s
10. Retrieve all employees in department 5 whose salary is between 50,000 and 60,000(inclusive)
11. Retrieve the names of all employees who do not have supervisors
12. Retrieve SSN and department name for all employees
13. Retrieve the name and address of all employees who work for the 'Research' department
14. For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.
15. For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.
16. Retrieve all combinations of Employee Name and Department Name
17. Make a list of all project numbers for projects that involve an employee whose last name is 'Narayan' either as a worker or as a manager of the department that controls the project.
18. Increase the salary of all employees working on the 'ProductX' project by 15%. Retrieve employee name and increased salary of these employees.
19. Retrieve a list of employees and the project name each works in, ordered by the employee's department, and within each department ordered alphabetically by employee first name.
20. Select the names of employees whose salary does not match with salary of any employee in department 10.
21. Retrieve the employee numbers of all employees who work on project located in Bellaire, Houston, or Stafford.
22. Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary. Display with proper headings.
23. Find the sum of the salaries and number of employees of all employees of the 'Marketing' department, as well as the maximum salary, the minimum salary, and the average salary in this department.
24. Select the names of employees whose salary is greater than the average salary of all employees in department 10.
25. Delete all dependents of employee whose **ssn is '123456789'**.
26. Perform a query using alter command to drop/add field and a constraint in Employee table.

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**II B.Sc. – IV Semester**

**Course: OBJECT ORIENTED PROGRAMMING THROUGH JAVA**

**Course Code:**

**No. of Hours/Week: 4**

**Paper : IV**

**Course Objective:**

To impart knowledge on fundamentals of Object Oriented Programming, classes, inheritance, interfaces and packages and to make the students understand the concept of exception handling and multithreading.

**Course Outcomes:**

At the end of the course the student will be able to

1. Understand and Apply Object Oriented features and understand the basics of Java.
2. Develop problem-solving and programming skills using OOP concepts.
3. Apply the concepts of inheritance and to create arrays, strings.
4. Able to demonstrate Exception Handling and Multithreading.
5. Develop efficient Java applets and applications using OOP concepts.

**UNIT- I**

**Introduction to Java:** Features of Java, The Java virtual Machine, Parts of Java

**Naming Conventions and Data Types:** Naming Conventions in Java, Data Types in Java, Literals

**Operators in Java:** Operators, Priority of Operators

**Control Statements in Java:** if... else Statement, do... while Statement, while Loop, for Loop, switch Statement, break Statement, continue Statement, return Statement

**Input and Output:** Accepting Input from the Keyboard, Reading Input with Java.util.Scanner Class, Displaying Output with System.out.printf(), Displaying Formatted Output with String.format()

**Arrays:** Types of Arrays, Three Dimensional Arrays (3D array), arrayname.length, Command Line Arguments

**UNIT-II**

**Strings:** Creating Strings, String Class Methods, String Comparison, Immutability of Strings

**Introduction to OOPs:** Problems in Procedure Oriented Approach, Features of Object-Oriented Programming System (OOPS)

**Classes and Objects:** Object Creation, Initializing the Instance Variables, Access Specifiers, Constructors

**Methods in Java:** Method Header or Method Prototype, Method Body, Understanding Methods, Static Methods, Static Block, The keyword 'this', Instance Methods, Passing Primitive Data Types to Methods, Passing Objects to Methods, Passing Arrays to Methods, Recursion, Factory Methods

**Inheritance:** Inheritance, The keyword 'super', The Protected Specifier, Types of Inheritance

### UNIT-III

**Polymorphism:** Polymorphism with Variables, Polymorphism using Methods, Polymorphism with Static Methods, Polymorphism with Private Methods, Polymorphism with Final Methods, final Class

**Type Casting:** Types of Data Types, Casting Primitive Data Types, Casting Referenced Data Types, The Object Class , **Abstract Classes:** Abstract Method and Abstract Class

**Interfaces:** Interface, Multiple Inheritance using Interfaces

**Packages:** Package, Different Types of Packages, The JAR Files, Interfaces in a Package, Creating Sub Package in a Package, Access Specifiers in Java, Creating API Document

**Exception Handling:** Errors in Java Program, Exceptions, throws Clause, throw Clause, Types of Exceptions, Re – throwing an Exception

### UNIT-IV

**Streams:** Stream, Creating a File using FileOutputStream, Reading Data from a File using FileInputStream, Creating a File using FileWriter, Reading a File using FileReader, Zipping and Unzipping Files, Serialization of Objects, Counting Number of Characters in a File, File Copy, File Class

**Threads:** Single Tasking, Multi Tasking, Uses of Threads, Creating a Thread and Running it, Terminating the Thread, Single Tasking Using a Thread, Multi Tasking Using Threads, Multiple Threads Acting on Single Object, Thread Class Methods, Deadlock of Threads, Thread Communication, Thread Priorities, thread Group, Daemon Threads, Applications of Threads, Thread Life Cycle

### UNIT-V

**Applets:** Creating an Applet, Uses of Applets, <APPLET> tag, A Simple Applet, An Applet with Swing Components, Animation in Applets, A Simple Game with an Applet, Applet Parameters

**Java Database Connectivity:** Database Servers, Database Clients, JDBC (Java Database Connectivity), Working with Oracle Database, Working with MySQL Database, Stages in a JDBC Program, Registering the Driver, Connecting to a Database, Preparing SQL Statements, Using jdbc–odbc Bridge Driver to Connect to Oracle Database, Retrieving Data from MySQL Database, Retrieving Data from MS Access Database, Stored Procedures and CallableStatements, Types of Result Sets

#### **Additional Inputs:**

Vectors, Wrapper classes, Hiding classes, static Import, Event Handling in Java.

#### **Text Books:**

1. Java - The Complete Reference by Herbert Schildt, 9<sup>th</sup> Edition, Oreilly Publications.
2. Introduction to Java Programming, by Y Daniel Liang, Seventh Edition, Pearson, 2017.

#### **Reference Books:**

1. Programming with JAVA, A primer by E.Balaguruswamy, 3e, TATA McGraw-Hill Company.
2. Programming with Java by John R. Hubbard, Second Edition, Schaum's outline Series, TATA McGraw-Hill.
3. Java TM: How to Program by Deitel&Deitel, PHI (2007).
4. Java Programming: From Problem Analysis to Program Design by D.S Mallik.
5. Core Java: An Integrated Approach, Authored by Dr. R. Nageswara Rao & Kogent Learning Solutions Inc.

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**II B.Sc. – IV Semester**

**Course: OBJECT ORIENTED PROGRAMMING USING JAVA LAB**

**Course Code:**

**No. of Hours/Week: 2**

**Course Objective:**

To introduce various Object Oriented Concepts through which the students will be enabled to implement classes, inheritance, interface, package and multithreading concepts.

**Course Outcomes:**

At the end of the course the student will be able to

1. Apply OOP concepts to solve real time problems.
2. Make use of class, inheritance, interface and packages to develop solutions for complex problems.
3. Build java applications using Exception handling and Threads.

**List of Experiments**

1. Write a program to read *Student Name, Reg.No, Marks[5]* and calculate *Total, Percentage, Result*. Display all the details of students
2. Write a program to perform the following String Operations
  - a. Read a string
  - b. Find out whether there is a given substring or not
  - c. Compare existing string by another string and display status
  - d. Replace existing string character with another character
  - e. Count number of words in a string
3. Java program to implements Addition and Multiplication of two N X N matrices.
4. Java program to demonstrate the use of Constructor.
5. Calculate area of the following shapes using method overloading.
  - a. Triangle
  - b. Rectangle
  - c. Circle
  - d. Square
6. Implement inheritance between *Person (Aadhar, Surname, Name, DOB, and Age)* and *Student (Admission Number, College, Course, Year)* classes where *ReadData(), DisplayData()* are overriding methods.
7. Java program for implementing Interfaces
8. Java program on Multiple Inheritance.
9. Java program for to display *Serial Number from 1 to N* by creating two Threads
10. Java program to demonstrate the following exception handlings
  - a. Divided by Zero
  - b. Array Index Out of Bound

- c. File Not Found
  - d. Arithmetic Exception
  - e. User Defined Exception
11. Create an Applet to display different shapes such as Circle, Oval, Rectangle, Square and Triangle.
12. Write a program to create **Book (ISBN, Title, Author, Price, Pages, Publisher)** structure and store book details in a file and perform the following operations
- a. Add book details
  - b. Search a book details for a given ISBN and display book details, if available
  - c. Update a book details using ISBN
  - d. Delete book details for a given ISBN and display list of remaining Books



**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**II B.Sc. – IV Semester**

**Course: OPERATING SYSTEMS**

**Course Code:**

**No. of Hours/Week: 4**

**Paper : V**

**Course Objective:**

To provide knowledge about the services and functions rendered by operating systems and inculcate knowledge on Process Scheduling and Memory Management.

**Course Outcomes:**

At the end of the course the student will be able to

1. Interpret the basic structure of OS and architectural components.
2. Compare and contrast various Process scheduling algorithms.
3. Analyze various mechanisms of Synchronization and the principles of deadlock.
4. Make use of paging and segmentation in Memory management.
5. Discuss the issues related to file system interface, implementation and disk management.

**UNIT - I**

What is Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

**UNIT - II**

Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Preemptive and Preemptive Scheduling Algorithms.

**UNIT - III**

**Process Management:** Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.

**Concurrent and Dependent Processes,** Critical Section, Semaphores, Methods for Inter-process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

**UNIT - IV**

**Memory Management:** Physical and Virtual Address Space; Memory Allocation Strategies– Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory.

**UNIT - V**

**File and I/O Management, OSsecurity** : DirectoryStructure, File Operations, File Allocation Methods, Device Management, Pipes, Buffer, Shared Memory, Security Policy Mechanism, Protection, Authentication and Internal Access Authorization  
Introduction to Android Operating System, Android Development Framework, Android Application Architecture, Android Process Management and File System, Small Application Development using Android Development Framework.

**Additional Inputs:**

Inter process Communication, Inter process Communication in Linux, Multithreading: Benefits, Multithreading Models.

**Text Books:**

1. Operating System Concepts - Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, 9<sup>th</sup> Edition, John Wiley and Sons Inc., 2012.
2. Operating Systems - Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2011.

**Reference Books:**

1. Modern Operating Systems, Andrew S. Tanenbaum, Second Edition, Addison Wesley, 2001.
2. Operating Systems: A Design-Oriented Approach, Charles Crowley, Tata McGraw Hill Education”, 1996.
3. Operating Systems: A Concept-Based Approach, D M Dhamdhare, Second Edition, Tata McGraw-Hill Education, 2007.
4. Operating Systems by J. Archer Harris (Author), Jyoti Singh (Author) (TMH)
5. Online Resources for UNIT V

**A.S.D. GOVERNMENT DEGREE COLLEGE FOR WOMEN(A)**  
**DEPARTMENT OF COMPUTER SCIENCE**  
**II B.Sc. – IV Semester**

**Course: OPERATING SYSTEMS LAB USING C/JAVA**

**Course Code:**

**No. of Hours/Week: 2**

**Course Objective:**

To impart knowledge on Operating System design Principles and demonstrate Process Scheduling and Page Replacement Algorithms.

**Course Outcomes:**

At the end of the course the student will be able to

1. Implement Process Scheduling and Page Replacement Algorithms.
2. Implement Various File Organization schemes
3. Implement Deadlock Avoidance and prevention algorithms

**List of Experiments**

1. Write a program to implement Round Robin CPU Scheduling algorithm
2. Simulate SJF CPU Scheduling algorithm
3. Write a program the FCFS CPU Scheduling algorithm
4. Write a program to Priority CPU Scheduling algorithm
5. Simulate Sequential file allocation strategies
6. Simulate Indexed file allocation strategies
7. Simulate Linked file allocation strategies
8. Simulate MVT and MFT memory management techniques
9. Simulate Single level directory File organization techniques
10. Simulate Two level File organization techniques
11. Simulate Hierarchical File organization techniques
13. Write a program for Bankers Algorithm for Dead Lock Avoidance
14. Implement Bankers Algorithm Dead Lock Prevention.
15. Simulate all Page replacement algorithms.
  - a. FIFO
  - b. LRU
  - c. LFU
16. Simulate Paging Techniques of memory management