BCA Syllabus Grid 2025-26

SEMESTER	SUBJECT
I	Computer Fundamentals and Office Automation(T) Computer Fundamentals and Office Automation(P) Problem Solving using C (T) Problem Solving using C (P)
II	Data Structures using C (T) Data Structures using C (P) Database Management System(T) Database Management System(P)
III	Database Management System(T) Database Management System(P) Data Structures(T) Data Structures(P) Object Oriented Programming Through JAVA(T) Object Oriented Programming Through JAVA(P) Software Engineering(T) Software Engineering(P)
IV	Python Programming(T) Python Programming(P) Operating Systems(T) Operating Systems(P) Mobile Application Development using Android(T) Mobile Application Development using Android(P)

ST. JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

I SEMESTER BCA HONORS Time: 3Hrs/Week

BCA-Ma1-1101 COMPUTER FUNDAMENTALS AND OFFICE AUTOMATION Max.Marks:100

Course Objectives:

1. **Understand foundational computing concepts**, including number systems, the evolution of computers, block diagrams, and generational progress.
2. **Develop knowledge of computer architecture**, focusing on system organization and networking fundamentals.
3. **Acquire practical skills in document creation**, formatting, and digital presentations using word processing tools.
4. **Gain proficiency in spreadsheet operations**, such as data entry, formulas, functions, and charting techniques.
5. **Introduce data visualization and basic modelling principles**, fostering analytical thinking in structuring and interpreting data sets.

Course Outcomes:

1. At the End of the Course, The Students will be able to **Explain** different number systems, the historical evolution of computers, and identify key components in a block diagram.

2. Learners will demonstrate **basic blocks of a computer and fundamental networking knowledge**.

3. Learners will create professional-level documents and **design visually appealing presentations** using word processing software and presentation software.

4. Learners will manipulate data within spreadsheets, apply formulas, and **generate accurate summaries and visualizations**.

5. Learners will apply data modelling techniques to **analyze, organize, and represent data effectively** in various scenarios**.**

Unit 1. Number Systems, Evolution, Block Diagram and Generations:

Number Systems: Binary, Decimal, Octal, Hexadecimal; conversions between number systems.

Evolution of Computers: History from early mechanical devices to modern-day systems.

Block Diagram of a Computer: Components like Input Unit, Output Unit, Memory, CPU (ALU + CU). **Generations of Computers**: First to Fifth Generation – technologies, characteristics, examples.

Unit 2. Basic organization and N/W fundamentals:

Computer Organization: Functional components – Input/Output devices, Storage types, Memory hierarchy.

Types of Computers: Micro, Mini, Mainframe, and Supercomputers.

Networking Fundamentals: Definition, need for networks, types (LAN, WAN, MAN), topology

(Star, Ring, Bus).

Internet Basics: IP Address, Domain Name, Web Browser, Email, WWW.

Unit 3. Word Processing and presentations:

Word Processing Basics: Using MS Word/Google Docs – formatting, styles, tables, mail merge.

Presentation Tools: Using PowerPoint/Google Slides – slide design, animations, transitions.

Applications: Creating resumes, reports, brochures, and presentations.

Keyboard Shortcuts

Unit 4. Spreadsheet Basics:

Spreadsheet Concepts: Understanding rows, columns, cells in tools like MS Excel/Google
Sheets, cell referencing.

Functions and Formulae: SUM, AVERAGE, IF, COUNT.

Charts and Graphs: Creating visual representations

Data Handling: Sorting, filtering, conditional formatting.

Text Functions: LEFT, RIGHT, MID, LEN, TRIM, CONCAT, TEXTJOIN

Advanced Functions: Logical: IF, AND, OR, IFERROR, **Lookup:** VLOOKUP, HLOOKUP, XLOOKUP, INDEX, MATCH

Unit 5. Data Analysis and Visualization:

Conditional Formatting: Custom rules, Color scales, Icon sets, Data bars

Data Analysis Tools: Pivot Tables and Pivot Charts, Data Validation (Drop-downs, Input Messages, Error Alerts), What-If Analysis: Goal Seek, Scenario Manager, Data Tables

Charts and Dashboards: Creating Interactive Dashboards, Using slicers with Pivot Tables, Combo Charts and Sparklines

Productivity Tips: Using Named Ranges, Freeze Panes, Split View

Textbooks:

1. Fundamentals of Computers, Reema Thareja, Oxford University Press, Second Edition
2. Fundamentals of Computers, V. Rajaraman – PHI Learning
3. Introduction to Computers by Peter Norton – McGraw Hill
4. Microsoft Office 365 In Practice by Randy Nordell – McGraw Hill Education

References:

1. Excel 2021 Bible by Michael Alexander, Richard Kusleika – Wiley
2. Networking All-in-One For Dummies by Doug Lowe – Wiley
3. Microsoft Official Docs and Training: https://learn.microsoft.com
4. Google Workspace Learning Center: https://support.google.com/a/users/

Activities:

Outcome: At the End of the Course, The Students will be able to explain different number systems, the historical evolution of computers, and identify key components in a block diagram.

Activity: Create a digital poster or infographic comparing number systems (binary, decimal, octal, hexadecimal) and illustrating the timeline of computer generations with key innovations.

Evaluation Method: Rubric-based assessment of the poster presentation on a 10-point scale

focusing on:

- Accuracy of number system conversions
- Visual organization and creativity
- Correct identification of block diagram components

Outcome: Learners will demonstrate basic blocks of a computer and fundamental networking

knowledge.

Activity: Design a concept map showing the internal architecture of a computer and types of

networks (LAN, WAN, MAN), including devices and topologies.

Evaluation Method: Checklist-based peer review and instructor validation:

- Completeness of the map
- Correctness of networking concepts
- Use of appropriate terminology
- Logical flow and structure of the map

Outcome: Learners will create professional-level documents and design visually appealing presentations using word processing software and presentation software.

Activity: Prepare a formal report (e.g., project proposal) in a word processor and present it using a slide deck with transitions, embedded media, and design elements.

Evaluation Method: Performance-based evaluation using a 10-point scoring scale:

- Formatting and structure of the document
- Presentation aesthetics and clarity
- Communication skills during presentation

Outcome: Learners will manipulate data within spreadsheets, apply formulas, and generate accurate summaries and visualizations.

Activity: Analyze a dataset (e.g., student scores or sales data) using spreadsheet software. Apply formulas (SUM, AVERAGE, IF, VLOOKUP) and create relevant charts.

Evaluation Method: Practical test with a rubric:

- Correct use of formulas
- Accuracy of data summaries

Outcome: Learners will apply data modelling techniques to analyze, organize, and represent data effectively in various scenarios.

Activity: Prepare an interactive dashboard for a given data set using EXCEL.

Evaluation Method: Evaluation of the dashboard on a 10-point scoring scale:

- Presentation aesthetics and clarity
- Interactiveness
- Communication skills during presentation

ST. JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

I SEMESTER **BCA HONORS** **Time: 2Hrs/Week**

BCA-Ma1-1151 COMPUTER FUNDAMENTALS AND OFFICE AUTOMATION Max.Marks:50

PRACTICAL

Course Objectives:

1. **Acquire practical skills in document creation**, formatting, and digital presentations using word processing tools.
2. **Gain proficiency in spreadsheet operations**, such as data entry, formulas, functions, and charting techniques.
3. **Introduce data visualization and basic modelling principles**, fostering analytical thinking in structuring and interpreting data sets.

Course Outcomes:

At the End of the Course, The Students will be able to:

1. Create professional-level documents and **design visually appealing presentations** using word processing software and presentation software.
2. Manipulate data within spreadsheets, apply formulas, and **generate accurate summaries and visualizations**.
3. Apply data modelling techniques to **analyze, organize, and represent data effectively** in various scenarios**.**

List of Experiments:

1. Demonstration of Assembling and Dessembling of Computer Systems.
2. Identify and prepare notes on the type of Network topology of your institution.
3. Prepare your resume in Word.
4. Using Word, write a letter to your higher official seeking 10-days leave.
5. Prepare a presentation that contains text, audio and video.
6. Using a spreadsheet, prepare your class Time Table.
7. Using a Spreadsheet, calculate the Gross and Net salary of employees(Min 5) considering
all the allowances.
8. Generate the class-wise and subject-wise results for a class of 20 students. Also generate
the highest and lowest marks in each subject.
9. Using IF, AND, OR, and IFERROR to Automate Grade Evaluation.
    a. Create a table of student scores in different subjects.
    b. Use IF to assign grades (A/B/C/Fail).
    c. Use IFERROR to handle missing scores or invalid data.
10. Employee Database Search Using VLOOKUP, HLOOKUP, XLOOKUP, INDEX, and MATCH
    a. Create a database of employees (Name, ID, Department, Salary).
    b. Implement VLOOKUP to search by employee ID.
    c. Use HLOOKUP to extract department heads by role.
    d. Apply XLOOKUP for more flexible searches.

e. Use INDEX + MATCH as an alternative to VLOOKUP.

11. Sales Report Analysis Using Pivot Tables and Charts
    a. Use a dataset of product sales (Product, Region, Date, Quantity, Revenue).
    b. Create Pivot Tables to summarize data by region/product.
    c. Insert Pivot Charts for visual analysis (e.g., bar, line).
    d. Add slicers to make the dashboard interactive.
12. Designing a Data Entry Form with Drop-downs and Input Rules
    a. Create a student registration form.
    b. Add drop-down lists for course selection using Data Validation.
    c. Add input messages to guide users.
    d. Add error alerts for wrong entries.
13. Monthly Budget Planning using Goal Seek and Scenario Manager
    a. Create a simple personal budget (income, expenses, savings).
    b. Use Goal Seek to determine income needed to save a desired amount.
    c. Use Scenario Manager to compare different budgeting scenarios (best/ worst/ realistic case).
    d. Create a one-variable Data Table to analyze how different expenses affect savings.
14. Dashboard Creation Using Combo Charts, Sparklines & Slicers
    a. Use existing sales or attendance data.
    b. Insert combo charts (e.g., column + line).
    c. Add sparklines to show trends.
    d. Use slicers with Pivot Tables to control dashboard elements.
    e. Finalize and format for interactivity.

ST.JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

I SEMESTER BCA HONORS Time: 3Hrs/Week

BCA-Ma2-1201 PROBLEM SOLVING USING C Max.Marks:100

Course Objectives:

1. Understand the fundamentals of computer programming, Apply structured problem solving approaches using algorithms, flowcharts, and C programming constructs.
2. Develop efficient logic using decision-making, loop, and jump control statements.
3. Utilize derived data types like arrays and strings for modular program design.
4. Design and implement modular solutions using functions, recursive logic, pointer operations, and dynamic memory management.
5. Handle complex data structures including structures, unions, and text file operations.

Course Outcomes:

At the end of the course, students will be able to:

1. Understand basic computing concepts, programming paradigms and write structured C

programs.

2. Apply control flow statements to solve logical and repetitive tasks in C.
3. Implement arrays and string operations to manage and manipulate data efficiently.
4. Design modular code using functions, recursion, and appropriate parameter passing.
5. Utilize pointers and memory operations for effective data handling. Demonstrate competence in dynamic memory allocation and text file processing.

Unit 1. Introduction to computer programming:

Introduction, Types of software, Compiler and interpreter, Concepts of Machine level, Assembly level and high-level programming, Flowcharts and Algorithms, Fundamentals of C: History of C, Features of C, C Tokens-variables and keywords and identifiers, constants and Data types, Rules for constructing variable names, Operators, Structure of C program, Input /output statements in C - Formatted and Unformatted I/O

Unit 2. Control statements:

Decision making statements: if, if else, else if ladder, switch statements.

Loop control statements: while loop, for loop and do-while loop. Jump Control statements: break,continue and goto.

Unit 3. Derived data types in C:

Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays -Declaration, Initialization and Memory representation. Strings: Declaring & Initializing string variables; String handling functions, Character handling functions

Unit 4. Functions:

Pointers: Pointer data type, Pointer declaration, initialization, accessing values using pointers. Pointer arithmetic, Pointers and arrays.

Function Prototype, definition and calling. Return statement. Nesting of functions. Categories of functions. Recursion (Basic Concept only). Parameter Passing by address & by value. Local and Global variables.

Storage classes: automatic, external, static and register.

Unit 5. Dynamic Memory Management:

Introduction, Functions-malloc, calloc, realloc, free

Structures: Basics of structure, structure members, accessing structure members, nested structures, array of structures, structure and functions, structures and pointers.

Unions - Union definition; difference between Structures and Unions.

Working with text files - modes: opening, reading, writing and closing text files.

Text Books:

1. Programming in ANSI C, E. Balagurusamy, Tata McGraw Hill, 6 th Edn,
2. Computer fundamentals and programming in C, Reema Theraja, Oxford University Press

Reference Books:

1. Let us C, Y Kanetkar, BPB publications
2. Head First C: A Brain-Friendly Guide, David Griffiths, Dawn Griffiths

Activities:

Outcome: Understand basic computing concepts, programming paradigms and write structured C programs.

Activity: Create a concept map of computing fundamentals and programming paradigms

(procedural, structured, object-oriented). Then, they write a structured C program (e.g., a

calculator or student grade system) using proper syntax, indentation, and modular design.

Evaluation Method: Rubric-based Code Review & Viva to check the

- ○ The correctness of the concept map
- ○ Correct use of structure (main + functions)
- ○ Identification of paradigm used
- ○ Code readability and documentation

Outcome: Apply control flow statements to solve logical and repetitive tasks in C.

Activity: Implement a program that solves a logic puzzle (e.g., number guessing game, pattern generation, or prime number finder) using if, switch, for, while, and do-while.

Evaluation Method: Automated Test Cases + Peer Review to check the

○ Correct use of control statements
○ Logical correctness of output
○ Efficiency and edge case handling
○ Peer feedback on clarity and logic

Outcome: Implement arrays and string operations to manage and manipulate data efficiently.

Activity: Build a program that stores and arranges student marks in ascending and descending order using arrays and performs string operations like concatenation, comparing, and formatting names.

Evaluation Method: Functional Demonstration + Code Walkthrough to check the
○ Correct array and string usage
○ Memory efficiency
○ Handling of invalid inputs
○ Explanation of sorting/searching logic

Activity:

● Recursive Problem Solver

Students write a modular program to solve a recursive problem (e.g., factorial, Fibonacci,
or Tower of Hanoi) using functions with parameters and return values.

Evaluation Method:

● Code Trace + Written Quiz
○ Correct function decomposition
○ Proper parameter passing (by value/reference)
○ Recursion depth and base case handling
○ Quiz on tracing recursive calls

Outcome: Utilize pointers and memory operations for effective data handling. Demonstrate
competence in dynamic memory allocation and text file processing.

Activity: Create a program that dynamically stores user input (e.g., survey responses) using
pointers and writes/reads the data to/from a text file.

Evaluation Method: Memory Debugging + File I/O Assessment to check the
○ Proper use of malloc, calloc, realloc, and free
○ Pointer arithmetic and dereferencing
○ File creation, reading, writing, and error handling

ST. JOSEPH'S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

I SEMESTER BCA HONORS Time: 2Hrs/Week

BCA-Ma2-1251 PROBLEM SOLVING USING C LAB Max.Marks:50

Course Objectives:

1. Develop efficient logic using decision-making, loop, and jump control statements.
2. Utilize derived data types like arrays and strings for modular program design.
3. Design and implement modular solutions using functions, recursive logic, pointer operations, and dynamic memory management.
4. Handle complex data structures including structures, unions, and text file operations.

Course Outcomes:

At the end of the course, students will be able to:

1. Apply control flow statements to solve logical and repetitive tasks in C.
2. Implement arrays and string operations to manage and manipulate data efficiently.
3. Design modular code using functions, recursion, and appropriate parameter passing.
4. Utilize pointers and memory operations for effective data handling.
5. Demonstrate competence in dynamic memory allocation and text file processing.

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 change in parameter values while swapping two integers
variables using Call by Value & Call by Address
6. Write a program to perform various string operations.
7. Write a program to search an element in a given list of values.
8. Write a program that uses functions to add two matrices.
9. Write a program to calculate factorial of given integer value using recursive functions
10. Write a program for multiplication of two N X N matrices.
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 read / write the data from / to a file.

14. Write a program to reverse the contents of a file and store in another 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 detail using ISBN

d. Delete book details for a given ISBN and display list of remaining Books

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam
II SEMESTER B.C.A. (HONORS) Time: 3Hrs/Week
BCA-Ma1-2102 DATA STRUCTURES USING C Max.Marks:100

COURSE OBJECTIVES:

The objective of the course is to make a student understand, implement data structures and also organize and manage data for efficient access.

COURSE OUTCOMES:

1. Understand the concept of Algorithms, analyze the time and space complexity of algorithms and representation of data using linked lists. (L1)
2. Implementation of data using linked lists and operations to be performed on linked lists (L2)
3. Implementation of stack and Queues using linked lists and applications of stacks and queues (L3)
4. Evaluate and apply various sorting and searching algorithms, including queues and priority queues.(L5).
5. Implement and analyze tree and graph structures for efficient data traversal and manipulation.(L4)

Unit 1. Basic Concepts:

Algorithm: Definition and characteristics, Complexity analysis: Space Complexity, Time Complexity, Asymptotic Notations.

Introduction to Data structures: Definition, Types of Data structures, Abstract Data Types (ADT), Introduction to Linked Lists, Representation of linked lists in Memory, Comparison between Linked List and Array.

Unit 2. Linked Lists:

Types of Linked Lists - Singly Linked list, Doubly Linked list, Circularly Singly Linked list, Circularly Doubly Linked list; Implementation of Single Linked List ADT: Creating a List, Traversing a linked list, Searching in linked list, Insertion and deletion into linked list (At first Node, Specified Position, Last node).

Unit 3. Stacks and Queues:

Introduction to stack ADT, Implementation of stacks using array and Linked List, Application of stacks - Polish Notations - Converting Infix to Post Fix Notation - Evaluation of Post Fix Notation.

Queues: Introduction to Queue ADT, Implementation of Queues using array and Linked List, Application of Queues Types of Queues- Circular Queues, De-queues, Priority Queue, Heaps.

Unit 4. Searching and Sorting:

Searchings: Linear or Sequential Search, Binary Search, Hashing and collision resolution.
Sorting: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort and Merge Sort

Unit 5. Trees and Graphs:

Trees: Tree Terminology, Binary Tree Representation, Traversal techniques, Expression Tree, Binary Search Tree- Definition, Operations on a Binary Search Tree: Creation, Search, Insertion & deletion.

Graphs: Introduction to Graphs, Terminology, Representation (Adjacency Matrix, Adjacency List), Traversal of Graphs (DFS, BFS), Applications of Graphs, Concept of Shortest Path Problems, Concept of Minimum Cost Spanning Tree

Textbooks:
1. Data Structures Using C, Balagurusamy E. Tata MCGraw Hill
2. Data Structures using C, Reema Thareja, Third Edition, Oxford University Press

Reference Books:
1. Data Structures, Lipschutz, Schaum's Outline Series, Tata Mcgraw-hill
2. Data Structures Using C, Ch. Vijay Kumar, Pen Press International

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam
II SEMESTER B.C.A. (HONORS) Time: 2Hrs/Week
BCA-Ma1-2152 DATA STRUCTURES USING C Max.Marks:50

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 values and index

d. Delete an existing element

2. Write a program to implement Single Linked List with insertion, deletion and traversal operations

3. Write a program to implement Doubly Linked List with insertion, deletion and traversal operations

4. Write a program to implement the Stack operations using Arrays and Linked Lists.

5. Write a program to convert a given infix expression to a postfix expression using stacks.

6. Write a program to implement the Queue operations using Arrays and Linked Lists.

7. Write a program to implement the Circular Queue operations using Arrays.

8. Write a program for Binary Search Tree Traversals

9. Write a program to search an item in a given list using the following Searching Algorithms

a. Linear Search

b. Binary Search.

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

a. Bubble Sort

b. Insertion Sort

c. Quick Sort

d. Merge Sort

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam
II SEMESTER B.C.A. (HONORS) Time: 3Hrs/Week
BCA-Ma2-2202 DATABASEMANAGEMENTSYSTEMS Max.Marks:100

COURSE OBJECTIVES:

A comprehensive understanding of database systems and their role in modern data management. Enable students to design, develop, and manage databases effectively using fundamental concepts and practical tools.

COURSE OUTCOMES:

Describe the fundamental concepts of databases, and database management systems, and limitations of file-based systems and DBMS, evaluate database system architectures, data models. (L1, L6)

3. **Construct** Entity-Relationship (ER) and Enhanced ER (EER) diagrams to **model** real-world scenarios, and **translate** them into relational schemas. **(L2)**
3. Apply relational algebra operations and analyze data dependencies to design normalized relational database schemas ensuring data integrity and consistency**. (L2)**
4. Develop SQL queries using DDL, DML, and DCL commands to retrieve, manipulate, and manage data efficiently in a relational database environment. (L3)
5. **Design and implement** database programs using PL/SQL, including control structures, loops, procedures, and functions to **enhance** database functionality.(L4).

Unit 1. 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 2. 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 modeling.

Unit 3. 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, Functional dependencies and normal forms.

Unit 4. Structured Query Language:

Introduction, 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 5. PL/SQL:

Introduction, Shortcomings of SQL, Structure of PL/SQL, PL/SQL Language Elements, Data Types, Operators Precedence, Control Structures, Steps to Create a PL/SQL, Program, Iterative Control, Procedures, Functions.

Textbooks:

1. Database System Concepts, Avi Silberschatz, Henry F. Korth,S. Sudarshan, Seventh

Edition, McGraw-Hill

2. Database Management Systems by Raghu Ramakrishnan, McGrawhill

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam
II SEMESTER B.C.A. (HONORS) Time: 2Hrs/Week
BCA-Ma2-2252 DATABASE MANAGEMENT SYSTEMS Max.Marks:50
List of Experiments

COURSE OBJECTIVES:

A comprehensive understanding of database systems and their role in modern data management. Enable students to design, develop, and manage databases effectively using fundamental concepts and practical tools.

COURSE OUTCOMES:

1. **Construct** Entity-Relationship (ER) and Enhanced ER (EER) diagrams to **model** real-world scenarios, and **translate** them into relational schemas. **(L2)**
2. Apply relational algebra operations and analyze data dependencies to design normalized relational database schemas ensuring data integrity and consistency**. (L2)**
3. Develop SQL queries using DDL, DML, and DCL commands to retrieve, manipulate, and manage data efficiently in a relational database environment. (L3)
4. **Design and implement** database programs using PL/SQL, including control structures, loops, procedures, and functions to **enhance** database functionality.(L4).

List of Experiments

Experiment 1 : Database: Inventory Management

Table 1: Products
Structure:

Column Name	Data Type	Constraints
product_id	INT	PRIMARY KEY
product_name	VARCHAR(50)	NOT NULL
price	DECIMAL(10,2)	CHECK(price > 0)
stock_qty	INT	CHECK(stock_qty>= 0)

Sample Data:

product_id	product_name	price	stock_qty
1	Pen	10.00	100
2	Notebook	50.00	200
3	Stapler	120.00	50
4	Marker	25.00	80

5	File Folder	60.00	150

Table 2: Suppliers Structure:

Column Name	Data Type	Constraints
supplier_id	INT	PRIMARY KEY
supplier_name	VARCHAR(50)	NOT NULL
contact_no	VARCHAR(20)	UNIQUE
product_id	INT	FOREIGN KEY REFERENCES Products(product_id)

Sample Data:

supplier_id	supplier_name	contact_no	product_id
101	StationeryMart	9876543210	1
102	PaperWorld	9876500000	2
103	OfficeSupplies	9876512345	3
104	MarkerHub	9876522222	4
105	FileDepot	9876533333	5

Section A: DDL (Data Definition Language)

1. Create a database called InventoryDB.
2. Create a table Products and table Suppliers with the specified columns and constraints:

Section B: DML (Data Manipulation Language)

4. Insert at least 5 rows into the Products table.
5. Insert at least 5 rows into the Suppliers table.
6. Update the stock quantity of product 'Pen' to 120.
7. Delete a supplier with a specific supplier_id.
8. Write a query to rename 'Notebook' to 'NoteBook A4'

Section C: DQL (SELECT Queries)

9. Display all records from the Products table.
10. Display only product_name and price of all products.
11. List all products that have a stock quantity less than 100.
12. Show all products between 20 and 100 price range.
13. Find all suppliers whose contact number starts with '98765'.
14. Find the average price of products.

15. Display the total number of products in the inventory.
16. Show the maximum and minimum stock quantities.
17. Count how many suppliers supply each product.
18. Show all products where price > 50 AND stock_qty> 100.
19. Show all products where price < 20 OR stock_qty< 80.
20. Display suppliers whose supplier_name contains the word 'Mart'
21. List all suppliers along with the product they supply (use INNER JOIN).
22. Display suppliers whose name starts with 'S'.
23. Find products whose name has exactly 5 characters
24. Find suppliers who supply products costing more than 100.

Experiment 2 : ONLINE BOOKSTORE DB

An online book store wants to implement a BOOKSTORE DB for managing their online transactions by using the following tables.

Authors Table

Column Name	Data Type	Constraints
author_id	INTEGER	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
nationality	VARCHAR	NULL allowed

Books Table

Column Name	Data Type	Constraints
book_id	INTEGER	PRIMARY KEY
Title	VARCHAR	NOT NULL
author_id	INTEGER	FOREIGN KEY REFERENCES Authors
publication_year	INTEGER	
Price	DECIMAL	

Customers Table

Column Name	Data Type	Constraints
customer_id	INTEGER	PRIMARY KEY
first_name	VARCHAR	NOT NULL

last_name	VARCHAR	NOT NULL
Email	VARCHAR	UNIQUE, NOT NULL
Address	VARCHAR	NOT NULL

Orders Table

Column Name	Data Type	Constraints
order_id	INTEGER	PRIMARY KEY
customer_id	INTEGER	FOREIGN KEY REFERENCES Customers
book_id	INTEGER	FOREIGN KEY REFERENCES Books
order_date	DATE	NOT NULL
quantity	INTEGER	NOT NULL

SAMPLE DATA SET for BOOKSTORE DB

Authors Table

author_id	first_name	last_name	nationality
1	Jane	Austen	British
2	George	Orwell	British
3	Gabriel	Garcia Marquez	Colombian
4	Toni	Morrison	American
5	Mark	Twain	American
6	Harper	Lee	American
7	Fyodor	Dostoevsky	Russian

Books Table

book_id	Title	author_id	publication_year	price
101	Pride and Prejudice	1	1813	12.99
102	1984	2	1949	9.50
103	One Hundred Years of Solitude	3	1967	15.00
104	Beloved	4	1987	11.25
105	Animal Farm	2	1945	8.75

106	Adventures of Huckleberry Finn	5	1884	10.50
107	To Kill a Mockingbird	6	1960	14.00

Customers Table

customer_id	first_name	last_name	Email	address
201	Alice	Smith	alice.s@example.com	12 Oak St, London
202	Bob	Johnson	bob.j@example.com	45 Pine Ave, Oxford
203	Charlie	Brown	charlie.b@example.com	78 Maple Rd, Bristol
204	Diana	Prince	diana.p@example.com	34 Queen St, York
205	Edward	Norton	edward.n@example.com	22 River Ln, Leeds
206	Fiona	Hall	fiona.h@example.com	56 Lake Dr, Bath
207	Greg	Miller	greg.m@example.com	89 Park Ave, Glasgow

Orders Table

order_id	customer_id	book_id	order_date	Quantity
301	201	101	2025-07-20	1
302	202	102	2025-07-21	2
303	201	105	2025-07-22	1
304	203	103	2025-07-23	1
305	204	106	2025-07-24	1
306	205	107	2025-07-25	3
307	206	104	2025-07-26	2

Section A: DDL (Schema Design & Constraints)

1. Write SQL statements to create all 4 tables (Authors, Books, Customers, Orders) with:

   o Primary Keys o Foreign Keys o Appropriate data types

   o NOT NULL constraints where necessary.

2. Alter the Books table to add a constraint that price must be greater than 0.

3. Add a new column phone_number to the Customers table (VARCHAR(15)) and ensure it is unique.

4. Drop the phone_number column from the Customers table.

Section B: DML (Data Manipulation)

5. Insert at least 7 records for each table (use sample dataset above).
6. Update the price of the book titled *Animal Farm* by increasing it by 10%.
7. Delete all orders made before 2025-07-21.
8. Change the nationality of Gabriel Garcia Marquez to "Latino-American".

Section C: SELECT Queries (Data Querying)

9. List all books published between 1900 and 2000.
10. Find all customers whose email contains "example.com".
11. Retrieve books whose price is between 10 and 15 and published before 1950.
12. Show authors who are either 'British' or 'American'.
13. Find books that have a price less than 10 or are published after 1980.
14. Display all orders placed after 2025-07-22.
15. List all books written by author with author_id = 2.
16. Find customers whose last name starts with B.
17. Show all books with a price NOT between 9 and 13.
18. Display books whose publication_year is in (1813, 1945, 1987).
19. Find authors whose nationality is NOT 'British'.
20. List customers whose address contains the word Park.
21. Show all books sorted by price in descending order.
22. List authors in alphabetical order by last_name.
23. Display orders sorted by order_date (latest first).

Use of Date Functions

24. Show all orders placed in July 2025.
25. Show all orders with an estimated delivery date (5 days after order date).
26. Show customers who placed an order on a weekend.
27. Calculate how many days have passed since the last order was placed.

Aggregate Functions (COUNT, SUM, AVG, MIN, MAX)

28. Count the total number of books in the database.
29. Find the average price of all books.
30. Show the highest-priced book.
31. Count how many orders each customer has placed.
32. Calculate the total sales (price × quantity) for each customer.

GROUP BY and HAVING

33. Count how many books are written by each author.
34. Group orders by customer_id and display total quantity ordered.
35. Show customers who have ordered more than 2 books in total (use HAVING).
36. Find the total number of books sold per author (GROUP BY author).

Experiment 3: EMPLOYEE DB

An enterprise wants to automate its employee management process by implementing an Employee Database.The goal is to replace manual record-keeping with a centralized system that stores employee, department, and project details. Use the following table structures and data set to implement Employee DB.

EmployeeDB – Table Structures

1. Departments Table

Column	**Type**	**Constraints**
dept_id	INT	PRIMARY KEY
dept_name	VARCHAR	UNIQUE, NOT NULL
location	VARCHAR	NOT NULL

Employees Table

Column	Type	Constraints
emp_id	INT	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
email	VARCHAR	UNIQUE, NOT NULL
phone	VARCHAR	CHECK (phone LIKE '--____')
hire_date	DATE	NOT NULL
job_title	VARCHAR	NOT NULL

salary	DECIMAL	CHECK (salary > 0)
dept_id	INT	FOREIGN KEY REFERENCES Departments(dept_id)
manager_id	INT	FOREIGN KEY REFERENCES Employees(emp_id) (selfreferential)

2. Projects Table

Column	Type	Constraints
project_id	INT	PRIMARY KEY
project_name	VARCHAR	NOT NULL
start_date	DATE	NOT NULL
end_date	DATE	NULL
dept_id	INT	FOREIGN KEY REFERENCES Departments(dept_id)

3. Employee_Project Table (Many-to-Many)

Column	Type	Constraints
emp_id	INT	FOREIGN KEY REFERENCES Employees(emp_id), PRIMARY KEY(emp_id, project_id)
project_id	INT	FOREIGN KEY REFERENCES Projects(project_id)
hours_allocated	INT	CHECK (hours_allocated> 0)

Sample Data Set

Departments Table

dept_id	dept_name	Location
1	HR	New York
2	IT	San Francisco
3	Finance	Chicago
4	Marketing	Boston
5	Operations	Seattle
6	Legal	Washington D.C.
7	Sales	Dallas

8	R&D	Austin
9	Procurement	Denver
10	Customer Care	Miami

2. Employees Table

emp_id	first_name	last_name	Email	phone	hire_date	job_title	salary	dept_id	manager_id
101	Alice	Johnson	alice.j@corp.com	123-4567890	2020-03-15	HR Manager	75000	1	NULL
102	Bob	Smith	bob.s@corp.com	234-5678901	2019-05-20	IT Analyst	65000	2	104
103	Charlie	Brown	charlie.b@corp.com	345-6789012	2021-01-10	Finance Executive	58000	3	106
104	Diana	Prince	diana.p@corp.com	456-7890123	2018-07-12	IT Manager	90000	2	NULL
105	Ethan	Hunt	ethan.h@corp.com	567-8901234	2022-02-25	Marketing Lead	62000	4	NULL
106	Fiona	Hall	fiona.h@corp.com	678-9012345	2017-11-01	Finance Manager	85000	3	NULL
107	Greg	Miles	greg.m@corp.com	789-0123456	2023-04-15	IT Support	45000	2	104
108	Hannah	White	hannah.w@corp.com	890-123-4567	2021-09-05	HR Executive	50000	1	101
109	Ian	Scott	ian.s@corp.com	901-2345678	2020-11-20	Operations Analyst	56000	5	NULL

110	Julia	Adams	julia.a@corp .com	012-3456789	2019-12-18	Legal Advisor	70000	6	NULL

3. Projects Table

project_id	project_name	start_date	end_date	dept_id
201	Payroll System	2023-01-01	NULL	3
202	Website Upgrade	2023-02-10	NULL	2
203	Recruitment Drive	2023-03-05	NULL	1
204	Ad Campaign	2023-05-20	NULL	4
205	New CRM Tool	2023-04-15	NULL	7
206	Compliance Portal	2023-06-10	NULL	6
207	Inventory System	2023-07-01	NULL	5
208	AI Research	2023-08-05	NULL	8
209	Customer Feedback	2023-09-10	NULL	10
210	Procurement System	2023-10-01	NULL	9

4. Employee_Project Table

emp_id	project_id	hours_allocated
102	202	120
104	202	80
103	201	100
106	201	150
101	203	50
105	204	70
107	202	60
109	207	90
110	206	110
108	203	40

Section A: DDL (Schema Creation & Modification)

1. Write SQL statements to create the above tables with the specified constraints 2. Alter the Employees table to add a column bonus DECIMAL(8,2) with default value0.

3. Drop the column bonus from Employees.

Section B: DML (Insert, Update, Delete)

3. Insert at least 10 rows into Departments, Employees, Projects, and Employee_Project.(use the above data set)
4. Try inserting an employee with a negative salary (should fail due to CHECK

   constraint).
5. Update the salary of the employee with emp_id = 103 by 15%.
6. Delete an employee record who has resigned (choose any emp_id).
7. Increase all employees' salaries in the IT department by 5%.
8. Change the department of an employee to "Research".(should fail due to FK

   constraint)

Section C: DQL (Select Queries)

9. List all employees and their details.
10. Show all employees in the "HR" department.
11. Find employees with salaries between 50,000 and 80,000.
12. Retrieve employees hired after 2020.
13. Show employees who are in either the IT or Finance department.
14. Find employees whose email ends with "@corp.com".
15. List all employees with salary > 60,000 AND located in "New York".
16. Display employees in descending order of salary.
17. Count the number of employees in each department.
18. Show the average salary of employees department-wise.
19. Display departments where the average salary is greater than 70,000.
20. Find the number of employees in each project.

21. Display departments with more than 3 employees.
22. Show the sum of all salaries department-wise.
23. List all distinct department IDs from the Employees table.
24. Show employee names with the year they were hired.
25. Show employees grouped by the year of hire.
26. List employees hired in the last 90 days.
27. List the no of years of experience of all the employees

Section D: Joins

28. List all employees with their department names (INNER JOIN).
29. Display all departments along with employees, including those departments without employees (LEFT JOIN).
30. Show employees and the projects they are working on (JOIN 3 tables: Employees, Employee_Project, Projects).
31. List projects along with total hours allocated by employees.
32. Write a query to find employees who are working on more than one project.
33. Show all projects handled by the 'Finance' department.

Section E: PL/SQL Programming

1. Write a procedure GetEmpInfo that takes emp_id as input and displays name, salary, and department.
2. Write a PL/SQL block that checks if an employee's salary is above 50,000. If yes, print "High Salary" ;Otherwise print "Standard Salary".
3. Write a PL/SQL program to display the top 10 rows in the Emp table based on their job and salary
4. Write a stored procedure GiveBonus that takes department ID and a designation as input, along with a bonus amount, and updates the salary of all employees in that department who have the specified designation by adding the bonus amount to their current salary.
5. Create a trigger to prevent inserting employees with a salary less than 30,000.
6. Create a trigger to avoid any transactions(insert, update, delete) on EMP table on Saturday & Sunday.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

III SEMESTER B.C.A. (HONORS) Time: 3Hrs/Week

BCA-Ma1-3101 DATABASE MANAGEMENT SYSTEM Max.Marks:100

Course Objectives:

- Graduates will have the expertise in analyzing real time problems and providing appropriate solutions related to Computer Science & Engineering.
- Graduates will have the knowledge of fundamental principles and innovative technologies to succeed in higher studies and research.
- Graduates will continue to learn and to adapt technology developments combined with deep awareness of ethical responsibilities in the profession.

Course Outcomes:

1. Analyze and design database systems using various models for data independence and efficiency. (L4)
2. Apply relational model principles and normalization techniques to design reliable schemas.(L3)
3. Design ER models and construct SQL queries for managing database schemas.(L6)
4. Construct complex SQL queries, manage relationships, and implement transaction and access controls for secure database operations.(L3)
5. Develop PL/SQL programs and implement transaction processing to ensure database reliability and integrity.(L3)

UNIT- I

Overview of Database Systems: Introduction: Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications.

Data Models: Introduction; types of data models, Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

Case Study:

1. Describe the differences between Database systems and File based systems

2. Study about database models and their advantages and dis-advantages

UNIT- II

Relational Model: Introduction to relational model, Codd's rules, concepts of domain, attribute, tuple, relation, constraints (Domain, Key constraints, integrity constraints) and their importance , concept of keys (super key, candidate key, primary key, surrogate key, foreign key) , relational Algebra & relational calculus.

Normalization: Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency(1NF, 2NF and 3 NF), Boyce-codd normal form(BCNF)

Case Study:

Describe Relational model and normalization for database design

UNIT - III:

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams,

BASIC SQL: Database schema, data types, DDL operations (create, alter, drop, rename), DML operations (insert, delete, update), basic SQL querying (select and project) using where clause, arithmetic & logical operations, aggregation, grouping, ordering.

Case Study:

1. Examine issues in data storage and query processing using SQL.
2. Create, maintain and manipulate a relational database using SQL

UNIT - IV

SQL: Nested queries/ sub queries, implementation of different types of joins, SQL functions(Date, Numeric, String, Conversion functions), Creating tables with relationship, implementation of key and integrity constraints, views, relational set operations , Transaction Control Language: commit, Rollback, Savepoint , DCL :Grant, Revoke

Case Study:

1. Try to convert some sample data to information and show how it can be used in decision making.

UNIT –V

PL/SQL: Introduction , Structure , Control Structures , Cursors , Procedure , Function , Packages , Exception Handling ,Triggers.

Transaction processing Concepts : Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Case Study:

Outline the role and issues in Transaction management of data such as efficiency, privacy, security.

Database management systems Text Books:

· Database Management Systems, 3rd Edition , Raghurama Krishnan, Johannes Gehrke, TMH · Database System Concepts,5th Edition , Silberschatz, Korth, TMH

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

III SEMESTER B.C.A. (HONORS) Time: 2hrs/Week

BCA-Ma1-3151 DATABASE MANAGEMENT SYSTEM

List of Experiments

SQL :

Cycle-I: Aim: Marketing company wishes to computerize their operations by using following tables.

Table Name: Client- Master

Description: Used to store client information

Column Name	Data Type	Size	Attribute
CLIENT_NO	Varchar2	6	Primary key
NAME	Varchar2	20	Not null
ADDRESS1	Varchar2	30	
ADDRESSS	Varchar2	30	
CITY	Varchar2	15	
PINCODE	Varchar2	8	
STATE	Varchar2	15	
BAL_DUE	Number	10, 2	

Table Name: Product_Master

Description: Used to store product information

ColumnName	Data Type	Size	Attribute
PRODUCT_NO	Varchar2	6	Primary Key
DESCRIPTION	Varchar2	15	Not null
PROFIT	Number	4,2	Not null

_PERCENT			
UNIT_MEASUE	Varchar2	10	
QTY_ON_HAND	Number	8	
REORDER_LVL	Number	8	
SELL_PRICE	Number	8,2	Not null, cannot be 0
COST _PRICE	Number	8,2	Not null,cannot be 0

Table Name: Salesman_master

Description: Used to store salesman information working for the company.

ColumnName	Data Type	Size	Attribute
SALESMAN_NO	Varchar2	6	Primary key
SALESMAN_NAME	Varchar2	20	Not null
ADDRESS1	Varchar2	30	
ADDRESS2	Varchar2	30	
CITY	Varchar2	20	
PINCODE	Number	8	
STATE	Vachar2	20	
SAL_AMT	Number	8,2	Not null, cannotbe0
TGT_TO_GET	Number	6,2	Not null, cannotbe0
YTD_SALES	Number	6,2	Not null
REMARKS	Varchar2	20	

Table Name: SALES

ORDER Description: Used to store client's orders

ColumnName	Data Type	Size	Attribute
ORDER_NO	Varchar 2	6	Primary Key
CLIENT_NO	Varchar 2	6	ForeignKey
ORDER _DATE	Date		
DELY_ADDRESS	Varchar 2	25	
SALESMAN_NO	Varchar 2	6	ForeignKey
DELY_TYPE	Char	1	Delivery:part(p)/full(f)and default iF'
BILL_YN	Char	1	
DELY_DATE	Date		Can't Be less than order date
ORDER _STATUS	Varchar 2	10	Values("InProcess","Fulfilled", "Back Order", "Canceled.

Table Name: SALES_ORDER_DETAILS

Description: Used to store client's order with details of each product ordered.

ColumnName	Data Type	Size	Attribute
ORDER_NO	Varchar 2	6	Primary key references SALES_ORDER table
PRODUCT_NO	Varchar 2	6	Foreign Key references SALES_ORDER_table
QTY_ ORDERED	Number	8	
QTY_ DISP	Number	8	

PRODUCT_RATE	Number	10,2	Foreign Key

Solve the following queries by using above tables.

1. Retrieve the list of names, city and the state of all the clients.

2. List all the clients who are located in 'Mumbai' or 'Bangalore'.

3. List the various products available from the product_master table.

4. Find the names of salesman who have a salary equal to Rs.3000.

5. List the names of all clients having 'a' as the second letter in their names.

6. List all clients whose Bal due is greater than value 1000.

7. List the clients who stay in a city whose first letter is 'M'.

8. List all information from sales-order table for orders placed in the month of July. 9. List the products whose selling price is greater than 1000 and less than or equal to 3000. 10. Find the products whose selling price is greater than 1000 and also find the new selling price as original selling price 0.50.

Cycle-II Supplier

Aim: A manufacturing company deals with various parts and various suppliers supply these parts. It consists of three tables to record its entire information. Those are as follows.

Supplier (Supplier_No, Sname, City, status) Part(Part_no, pname, color, weight, city, cost) Shipment (supplier_No, Part_no, city) JX(project_no, project_name, city)

SPJX(Supplier_no, part_no, project_no,city)

1. Get supplier numbers and status for suppliers in Chennai with status>20.

2. Get project names for projects supplied by supplier 'S'.

3. Get colors of parts supplied by supplier S'.

4. Get part numbers for parts supplied to any project in Mumbai.

5. Find the id's of suppliers who supply a red or pink parts.

Cycle–III EmployeeDatabase

Aim: An enterprise wishes to maintain a database to automate its operations. Enterprise divided into a certain departments and each department consists of employees. The following two tables describes the automation schemas.

Emp(Empno, Ename, Job, Mgr, Hiredate, Sal, Comm, Deptno)

Dept(Deptno, Dname, Loc)

1. List the details of employees who have joined before the end of September '81.

2. List the name of the employee and designation of the employee, who does not report to anybody.

3. List the name,salary and PF amount of all the employees(PF is calculated as10%of salary)

4. List the names of employees who are more than 2 years old in the organization.

5. Determine the number of employees, who are taking commission.

6. Update the employee salary by 20%,whose experience is greater than 12 years.

7. Determine the department does not contain any employees.

8. Create a view, which contains employee name and their manager names

working in sales department.

9. Determine the employees, whose total salary is like the minimum salary of any department.

10. List the department numbers and number of employees in each department.

PL/SQL PROGRAMS

1. Write a PL/SQL program to check if the given string is palindrome or not.

2. The HRD manager has decide to raise the employee salary by 15% write a PL/SQL block to accept the employee number and update the salary of that employee. Display appropriate message based on the existence of the record in Emp table.

3. Write a PL/SQL program to display top 10 rows in Emp table based on their job and salary.

4. Write a PL/SQL program to raise the employee salary by 10% for department number 30 people and also maintain the raised details in the raise table.

5. Create a procedure to update the salaries of Employees by 20%, for those who are not getting commission

6. Write a PL/SQL procedure to prepare an electricity bill by using following table. Table used: Elect

Name	Null?	Type
MNNO	NOT NULL	NUMBER(3)
CNAME		VARCHAR2(20)
CUR_READ		NUMBER(5)
PREV_READ		NUMBER(5)
NO_ UNITS		NUMBER(5)
AMOUNT		NUMBER(8,2)
SER_TAX		NUMBER(8,2)

NET_AMT		NUMBER(9,2)

7. Create a trigger to avoid any transactions(insert, update, delete) on EMP table on Saturday & Sunday.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

III SEMESTER B.C.A. (HONORS) Time: 3Hrs/Week

BCA-Ma2-3201 DATA STRUCTURES Max.Marks:100

Course Objectives:

- The objective of the course is to make a student implement data structures and organize and manage data, based on data structures for efficient access.

Course Outcomes:

The students will be able to

1. Analyze and evaluate data structures and algorithms to optimize computational efficiency.(L4)
2. Apply and manipulate arrays, records, and pointers to efficiently manage and access data structures.(L3)
3. Implement and analyze linked lists and stacks using dynamic memory allocation and various representations.(L4)
4. Evaluate and apply various sorting and searching algorithms, including queues and priority queues.(L5)
5. Implement and analyze tree and graph structures for efficient data traversal and manipulation.(L4)

Unit-I

Introduction and Overview- Elementary Data Organization, Data Structures classification, Data Structure Operations, Algorithms: Complexity, Time-Space Tradeoff.

Preliminaries-Mathematical Notation and Functions, Algorithmic Notation, Control Structures used in algorithms, Complexity of Algorithms. Other Asymptotic Notations, Sub algorithms, Variables, Data Types.

Case Study:

1. Calculate the space complexity of a given code

```
int tot (int a, int b)
{
        int c;
        c = a + b;
        return c;
}
```

Unit-II

Arrays, Records and Pointers – Linear Arrays, Representation and Traversing Linear Arrays, Inserting and Deleting. Passing an array to function, Pointer & Arrays Multidimensional Arrays, Sparse Matrices.

Case Study:

1. Application of arrays in the real world

Unit- III

Linked Lists – Representation, Dynamic Memory Allocation, Traversing, Searching, Insertion, Deletion, Header Linked Lists, Two-Way Lists

Stacks- Stacks, Operations on stacks, Array representation of stacks, Linked List representation of stacks, Arithmetic Expressions, Polish notation, Recursion.

Case Study:

1. Linked list verses Arrays.
2. Towers of Hanoi.

Unit- IV

Queues, Linked representation of Queues, Deques, Priority Queues.

Sorting - Insertion Sort, Bubble Sort, Selection sort, Quick Sort, Merge sort, Heap Sort, Searching – Linear Search, Binary Search.

Case Study:

1. Application of Queues.
2. Comparison of sorting algorithms.

Unit- V

Trees- Binary trees, Representing and traversing binary trees, Traversal algorithms using stacks. Binary Search Trees, Searching, Insertion and Deletion in Binary Search Trees,

Graphs- Terminology, Sequential representation of Graphs, Linked representation of Graphs, Operations on Graphs, Traversing a Graph.

Case Study:

1. Applications of Binary Tree.
2. Warshall's Algorithm.

Text books:

1. Data Structures by Seymour Lipschutz, McGraw Hill(Schaum's Outlines). 2. Data Structures using C , Second edition , Dr. Reema Thareja, Oxford University Press.

REFERENCE BOOKS:

1. Data Structures & Algorithms Using C, Khanna Publishers

2. Theory and Problems of Data Structures by Seymour Lipschutz, McGraw Hill (Schaum's Outlines)

3. Data Structures & Algorithms in C by M.A.Weiss, Addison Wisley.

4. Data Structures Using C, Reema Thareja, oxford.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

SEMESTER-III **B.C.A. (HONORS)** **Time: 2hrs/Week**

BCA-Ma2-3251 **DATA STRUCTURES LAB**

List of Lab Experiments

1. Write a C program to Implement matrix multiplication.
2. Write a C program to Implement stack using arrays.
3. Write a C program to Implement queue using arrays.
4. Write a C program to Implement circular queue using arrays.
5. Write a C program to Implement dequeue using arrays.
6. Write a C program to Implement single linked list using the methods create(), insert(), search(), delete() and display().
7. Write a C program to Implement double linked list.
8. Write a C program to Implement stack using linked list.
9. Write a C program to Implement queue using linked list.
10. Give a solution to towers of Hanoi using C program.
11. Write a C program to Implement bubble sort.
12. Write a C program to Implement selection sort.
13. Write a C program to Implement insertion sort.
14. Write a C program to Implement merge sort.
15. Write a C program to Implement quick sort.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

III SEMESTER B.C.A. (HONORS) Time: 3Hrs/Week

BCA-Ma3-3301 OBJECT ORIENTED PROGRAMMING THROUGH JAVA Max.Marks:100

Course Objectives:

- To make the students understand the fundamentals of Java programming.
- To expose the students to Window based applications using AWT
- To make the students to design appropriate Exception Handling in Java
- To make the students to understand the concepts of Threads Files and
- I/O Streams, Applets Networking in java.

Course Outcomes:

The student will be able to

1. Apply OOP concepts and Java fundamentals to design and implement Java applications.(L3)
2. Make use of control structures, classes, objects, and arrays to develop efficient Java code.(L3)
3. Analyze and apply inheritance, interfaces, and string operations in Java.(L3,L4)
4. Demonstrate packages and multithreading in Java.(L2)
5. Apply exception handling and applet programming in Java.(L3)

UNIT - I

Introduction to OOPS: Paradigms of Programming Languages – Basic concepts of Object Oriented Programming – Differences between Procedure Oriented Programming and Object Oriented programming - Benefits of OOPs – Application of OOPs. Java: History – Java features – Java Environment – JDK – API.

Introduction to Java: Creating and Executing a Java program – Java Tokens- Java Virtual Machine (JVM) – Command Line Arguments – Comments in Java program. Elements: Constants – Variables – Data types - Scope of variables – Type casting – Operators: Special operators – Expressions – Evaluation of Expressions.

Case Study:

1. Study the evolution of JAVA, why it was developed, and how it changed the software industry scenario.

UNIT - II

Control Structures: The if Statement, Nested ifs, The if-else-if Ladder and, Looping Statements: The while Loop, The do-while Loop, for loop and its variations and Nested Loops. Jumping Statements: Break, continue Statement.

Class and objects: Defining a class – Methods – Creating objects – Accessing class members – Constructors – Parameterized Constructors, Adding a Constructor. Arrays: One Dimensional Array – Creating an array – Array
processing – Multidimensional.

Case Study:

1. Study the difference between the looping structured in JAVA And Programming in C. 2. Study the limitation of Constructors in JAVA.

UNIT - III

Inheritance: Defining inheritance –types of inheritance– Method overloading – Static members – Nesting of Methods – this keyword - Overriding methods – Final variables and methods – Final classes – Final methods - Abstract methods and classes – Visibility Control.

Interfaces: Defining interface – Extending interface - Implementing Interface - Accessing interface variables. Strings: Constructing Strings, Operating on Strings, Arrays of Strings

Case Study:

1. Study the inheritance types available in JAVA and try to identify the limitations.

UNIT – IV

Packages: Java API Packages – Defining a Package, System Packages – Naming Conventions – Creating & Package Member Access – Adding Class to a Package.

Multithreading: Creating Threads – Life of a Thread – Defining & Running Thread – Thread Methods – Thread Priority – Synchronization –Implementing Runnable interface – Thread Scheduling.

Case Study:

1. Study the advantages of Package compared to Libraries in Procedural languages.

UNIT – V

Exception Handling: Limitations of Error handling – Advantages of Exception Handling - Types of Errors – Basics of Exception Handling - Syntax of Exception Handling Code, Multiple Catch Statements, Using finally Statement, Throwing Our Own Exceptions

Applets: Introduction, Java applications versus Java Applets, Applet Life-cycle, Working with Applets, The HTML Applet Tag.

Case Study:

1. Study and present the limitation of Applets in Web application development.

TEXT BOOKS:

1. **Object Oriented Programming through Java,** Universities Press, by P. Radha Krishna. 2. E. Balagurusamy, "*Programming with Java*", TataMc-Graw Hill, 5th Edition.

REFERENCES:

1. Herbert Schildt, "The complete reference Java", TataMc-Graw Hill, 7th Edition.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

III SEMESTER B.C.A. (HONORS) Time: 2hrs/Week

BCA-Ma3-3351 OBJECT ORIENTED PROGRAMMING THROUGH JAVA

List of Lab Experiments

1 Write a program to print the Biggest of 3 Numbers using Logical Operators.

2 Write a program to Test the Prime number.

3 Write a program to create a Simple class to find out the Area and perimeter of a rectangle and box using super and this keyword.

4 Write a program to design a class account using the inheritance and static that show all functions of the bank(withdrawal, deposit).

5 Write a program to design a class using abstract methods and classes.

6 Write a program to design a string class that performs the string method (equal, reverse the string, change case).

7 Write a program to handle the exception using try and multiple catch block.

8 Write a program that import the user-defined package and access the member variable of classes contained by the package.

9 Write a program that show the implementation of the interface.

10 Write a program to create a thread that implements the runnable interface.

11 Write a program to draw the line, rectangle, oval, text using the graphics method.

12 Write a program to create a menu using the frame.

13 Write a program to create a dialog box.

14 Write a program to implement the flow layout and border layout.

15 Write a program to create a Frame that display the student information.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam
III SEMESTER B.C.A. (HONORS) Time: 3Hrs/Week
BCA-Ma4-3401 SOFTWARE ENGINEERING Max.Marks:100

Course Objectives:

- The Objective of the course is to assist the student in understanding the basic theory of software engineering, and to apply these basic theoretical principles to a group software development project.

Course Outcomes

1. Categorize the specific requirements of the software projects.(L4)
2. Analyze software requirements with existing tools.(L4)
3. Identify various testing methodologies.(L3)
4. Apply the basic project management practices in real life projects.(L3)
5. Ability to work in a team as well as independently on software projects

UNIT I

Introduction to Software Engineering: Definitions - Size Factors - uality and Productivity Factors – Managerial Issues. Planning a software project: Defining the problem - Developing a Solution Strategy - Planning the Development Process - Planning an Organization structure - Other Planning Activities.

UNIT – II

Software Cost Estimation: Software cost factors - Software Cost.

Estimation Techniques – Staffing level Estimation- Estimating Software Maintenance Costs - The Software Requirements, Specification - Formal Specification Techniques - Languages and Processors for Requirements Specification.

UNIT – III

Software design: Fundamental Design Concepts - Modules and Modularization Criteria – Design Notations -Design Techniques - Detailed Design Considerations. Real-Time and Distributed System Design - Test Plans - Milestones, walkthroughs, and Inspections.

UNIT IV

User interface design and real time systems: User interface design - Human factors - Human computer interaction - Human - Computer Interface design - Interface design - Interface standards.

UNIT V

Software quality and testing: Software Quality Assurance - Quality metrics - Software Reliability - Software testing - Path testing – Control Structures testing - Black Box testing - Integration, Validation and system testing - Reverse Engineering and Reengineering.

CASE Tools: Projects management, tools - analysis and design tools – programming tools - integration and testing tool - Case studies.

REFERENCE BOOKS:

1. R.Fairley, Software Engineering Concepts, Tata McGraw-Hill, 1997.

2. R.S. Pressman, Software Engineering, Fourth Ed., McGraw Hill, 1997.

3. Software Engineering, H. Sommervill Ian , Addition Wesley Pub. Co.

4. Software Engineering: An object Oriented Perspective by Braude, E.J., Willey, 2001

Student Activity:

1. Visit any financial organization nearby and prepare requirement analysis report 2. Visit any industrial organization and prepare risk chart

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

III SEMESTER B.C.A. (HONORS) Time: 2hrs/Week

BCA-Ma4-3451 SOFTWARE ENGINEERING LAB

Case Studies:

1. Student Marks Analysis System
2. E-Commerce Management System
3. Inventory Control System
4. Food Delivery Management system
5. Logistics Management System

Choose any two of above case studies and do the following exercises for that Case Study

1. Write the complete problem statement
2. Write the software requirements specification document
3. Draw the entity relationship diagram
4. Draw the data flow diagrams
5. Draw use case diagrams
6. Draw activity diagrams for all use cases
7. Draw sequence diagrams for all use cases
8. Draw collaboration diagram
9. Assign objects in sequence diagrams to classes and make class diagram.

Note: 1. To draw dataflow diagrams using Microsoft Visio Software, SmartDraw, etc… 2. To draw UML diagrams using Rational Rose Software, Star UML, etc.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

IV SEMESTER B.C.A. (HONOURS) Time: 3Hrs/Week

BCA-Ma1-4101 PYTHON PROGRAMMING Max.Marks:100

Course Outcomes: The students will be able to

1. Recall fundamental Python concepts, including keywords, identifiers, variables, and string operations.[L1]
2. Apply functions, OOP principles, and exception handling to create and manage Python programs.[L3]
3. Utilize lists, tuples, and dictionaries to manipulate data in Python.[L3]
4. Analyze and manipulate data using NumPy and Pandas libraries in Python.[L4]
5. create visualizations with Matplotlib and develop GUI applications with database connectivity using Tkinter and MySQL.[L6]

Unit-I

Getting Started with Python: Introduction to Python , Python Keywords , Identifiers , Variables , Comments, Data Types , Operators, Input and Output , Type Conversion , Debugging . Flow of Control, Selection , Indentation , Repetition , Break and Continue Statement , Nested Loops

Strings- String Operations , Traversing a String , String handling Functions.

Case Study:

1. Study the features that make Python different from Procedural Languages.

Unit-II

Functions: Functions, Built-in Functions, User Defined Functions, recursive functions, Scope of a Variable

Python and OOP: Defining Classes, Defining and calling functions passing arguments, Inheritance, polymorphism, Modules – date time, math, Packages.

Exception Handling- Exception in python, Types of Exception, User-defined Exceptions.

Case Study:

1. Present a report of how Exception handling is different from JAVA Exceptional Handling.

Unit-III

List: Introduction to List, List Operations, Traversing a List, List Methods and Built-in Functions.

Tuples and Dictionaries, Introduction to Tuples, Tuple Operations, Tuple Methods and Built-in Functions, Nested Tuples. Introduction to Dictionaries, Dictionaries are Mutable, Dictionary Operations, Traversing a Dictionary, Dictionary Methods and Built-in functions.

Case Study:

1. What are the special features of dictionaries and try to analyze

about the same features in any other language.

Unit-IV

Introduction to NumPy, Array , NumPy Array , Indexing and Slicing , Operations on Arrays , Concatenating Arrays , Reshaping Arrays , Splitting Arrays , Statistical Operations on Arrays.

Data Handling using Pandas , Introduction to Python Libraries, Series, DataFrame, Importing and Exporting Data between CSV Files and DataFrames, Pandas Series Vs NumPy ndarray.

Case Study:

1. Present a paper on advanced features of NumPy and Pandas.

Unit-V

Plotting Data using Matplotlib: Introduction, Plotting using Matplotlib –Line chart, Bar chart, Histogram, Scatter Chart, Pie Chart.

GUI Programming and Database Connectivity Using Python. Graphical User Interfaces. Using the Tkinter Module, Creating Label, Text, Buttons, info Dialog Boxes, Radiobutton, Checkbutton, Getting Input, Importing MySQL for Python , Connecting with a database, Forming a query in MySQL, Passing a query to MySQL.

Case Study:

1. Present a paper on the features and advantages of MySQL compared to other commercial Databases.

References:

1. Mark Lutz, Learning Python,5th Ed. O"REILLY
2. Core Python Programming by Dr. R. Nageswara Rao
3. Problem Solving and Python Programming by E. Balaguru Swamy
4. Python programming: using problem solving approach by Reema Thareja. 5. Albert Lukaszewski ,MySQL for Python,Packet Publishing

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

IV SEMESTER B.C.A. (HONORS) Time: 2hrs/Week

BCA-Ma1-4151 PYTHON PROGRAMMING LAB

1. Write a Program to check whether given number is Armstrong or not.
2. Write a Program to check whether given number is perfect or not.
3. Write a program to find factorial of given number using recursive function
4. Write a program to implement inheritance and polymorphism
5. Demonstrate a python code to print try, except and finally block statements
6. Write a program to demonstrate String handling functions
7. Write a program to input n numbers from the user. Store these numbers in a tuple. Print the maximum and minimum number from this tuple.
8. Write a program to enter names of employees and their salaries as input and store them in a dictionary
9. Write a program to implement statistical operations on arrays using numPy
10. Write a program to import and export CSV file to DataFrame.
11. Create the DataFrame Sales containing year wise sales and perform basic operation on it.
12. Visualize the plots using matplot lib.
13. Create GUI interface with different types button and labels
14. Create GUI interface and connect with MySQL database and perform CRUD(Create, Read, Update and Delete) operations.

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

IV SEMESTER B.C.A. (HONOURS) Time: 3Hrs/Week

BCA-Ma2-4201 OPERATING SYSTEMS Max.Marks:100

Course Objectives:

1. To know the basic Structure, Components and Organization of Operating System.

2. To learn the notation of a Process- a Program in Execution, Management, Scheduling and Classic Problems of Synchronization.

3. To gain knowledge in various Memory Management Techniques.

4. To understand Unix Operating System and Various File operations.

Course Outcomes:

The students will be able to:

1. Recall the fundamental functions, services, and structures of operating systems, including UNIX.[L1]
2. Explain process management and scheduling concepts, including UNIX CPU scheduling.[L2]
3. Explain process synchronization, semaphore usage, and deadlock concepts, including handling approaches.[L4]
4. Analyze memory management concepts such as swapping, paging, and segmentation, with a focus on UNIX.[L4]
5. Identify UNIX file and directory management and compare it with Windows.[L3]

Unit I

Introduction: What is Operating System? ,History and Evolution of OS, Basic OS Functions, Computer System Architecture, Operating System Structure.

System Structures: Operating System Services, User Operating System Interface, System Calls, Types of System Calls, Overview of UNIX Operating System, Basic Features of Unix Operating System.

Case Study :

1. Understanding and listing the basic differences between UNIX OS and Windows OS in usage, user interface, features etc.

Unit II

Process Management: Process Concept, Operation on Processes, Communication in Client Server Systems.

Process Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, CPU Scheduling in UNIX.

Case Study:

1. Present your understanding on how CPU Scheduling is different in WINDOWS compared to UNIX/LINUX.

Unit III

Synchronization: Process Synchronization, Semaphores: Usage, Implementation, The Critical Section Problem., Classic problems of synchronization.

Deadlocks: Introduction, Deadlock Characterization, Necessary and Sufficient conditions for Deadlock, Deadlock Handling Approaches : Deadlock prevention, Deadlock Avoidance and Deadlock detection and Recovery .

Case Study:

1. Present your understanding of Deadlocks and new methodologies available in new Operating Systems released in the market.

Unit IV

Memory Management: Overview, Swapping, Contiguous Memory Allocation, Paging, Paging Examples, Segmentation, Page Replacement Algorithms, Memory management in UNIX.

Case Study:

1. Present a paper on new methods used in Memory management in the present day Operating Systems .

Unit V

Files and Directories in UNIX: Files, Directory Structure, File Operations, File System Implementation: File Allocation Methods, Comparison of UNIX and Windows.

Case Study:

1. Present a Paper on how UNIX treats regular files and directories differently from other operating systems.

TEXTBOOKS

1. Operating System Concepts: Abraham Silberschatz, Peter B. Galvin, GregGagne, 8th Edition,Wiley.
2. Unix and shell Programming by B.MH Arwani, OXFORD University Press.

REFERENCEBOOKS:

1. Operating System Principles, Abraham Silberchatz, PeterB.Galvin, GregGagne 8thEdition, WileyStudentEdition.
2. Principles of Operating Systems by Naresh Chauhan, OXFORD University Press.
3. Tanenbaum A S, Woodhull A S, Operating System Design and Implementation,3rd edition, PHI 2006.
4. Unix Shell Programming-Yashwant Kanetkar

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

IV SEMESTER **B.C.A. (HONORS)** **Time:** 2hrs/Week

BCA-Ma2-4251 **OPERATING SYSTEMS**

1. Introducing the LINUX Native editor vi: Working on basics of creating and editing a text file using standard commands of vi.

2. Introduction to UNIX Operating System, Compare with Windows OS. Writing and executing simple Hello World C Program in UNIX Environment.

3. Getting hands-on on basic UNIX Commands.

4. Write a program using the following system calls of UNIX OS fork, exec, getpid, exit, wait, close, opendir, readdir ?

5. Write a Simple shell script for basic arithmetic and logical calculations?

6. Write Shell script to check the given number is even or odd?

7. Write a shell script to swap the two integers?

8. Write Shell script to perform various operations on given strings.

9. Write Shell scripts to explore system variables such as PATH, HOME etc.

10. Write a shell script to display list of users currently logged in.

11. Write a shell script to delete all the temporary files.

12. Write a shell script to find the Factorial of a Number ?

13. Write C programs to implement the following Scheduling Algorithms: a) First Come First Serve.

b) Shortest Job First.

c) Round Robin.

Reference Text Books:

1. Brian W. Kernighan and Rob Pike, “The UNIX Programming Environment” Prentice Hall India (Edition available in LR Candin the form of E Book on student resource).

2. Yashwant Kanetkar, “UNIX Shell Programming” BPB Publications (First Edition).

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam

IV SEMESTER B.C.A. (HONOURS) Time: 3Hrs/Week

BCA-Ma3-4301 MOBILE APPLICATION DEVELOPMENT USING ANDROID Max.Marks:100

COURSE OBJECTIVES:

1. To facilitate students understanding android SDK
2. To help students to gain a basic understanding of Android application development 3. To instill working knowledge of Android Studio development tool

COURSE OUTCOMES:

The students will be able to

1. Demonstrate the Android platform, architecture, and application development process.[L2]
2. Apply Android design essentials to create UIs, manage activities, and handle intents and permissions.[L3]
3. Design Android UIs with screen elements, layouts, input controls, and custom lists.[L6]
4. Test, publish, and manage resources for Android applications.[L6]
5. Utilize common Android APIs to manage storage, databases, networking, location, and connectivity, and deploy applications.[L3]

UNIT-I

Introduction to Android: - Overview, History, Features of Android, The Android Platform, Understanding the Android Software Stack – Android Application Architecture –The Android Application Life Cycle – The Activity Life Cycle, Creating Android Activity -Views- Layout Android SDK, Android Installation, Building you First Android application, Understanding Anatomy of Android Application, Android Manifest file.

Case Study:

i. Give a brief description of Android Architecture and its parts.

ii. List out the challenges we face while using Android?

iii. List the new features of Android in the latest version.

UNIT-II

Android Application Design Essentials: Anatomy of an Android applications, Android terminologies, Creating User Interfaces with basic views- Application Context, Activities, Services, Intents, linking activities with Intents,, Receiving and Broadcasting Intents, Android Manifest File and its common settings, Using Intent Filter, Permissions.

Case Study:

i. Present an idea that you would like to convert it into an application in the future.

UNIT-III

Android User Interface Design Essentials: User Interface Screen elements, Designing User Interfaces with Layouts, Drawing and Working with Animation. Layouts, Recycler View, List View, Grid View and Web view

Input Controls: Buttons, Checkboxes, Radio Buttons, Toggle Buttons, Spinners, Input Events, Menus, Toast, Dialogs, Styles and Themes, Creating lists, and Custom lists

Case Study:

i. Present detail report on the features of Check Boxes, Radio Buttons and Toggle Buttons.

UNIT-IV

Testing Android applications: Publishing Android application, Using Android preferences, Managing Application resources in a hierarchy, working with different types of resources.

Case Study:

1. List out the special features of Android with its counterparts.

UNIT-V

Using Common Android APIs: Internal Storage, External Storage , SQLite Databases , Managing data using Sqlite, Sharing Data between Applications with Content Providers, Using Android Networking APIs, Using Android Web APIs, JSON Parsing, Using Android Telephony APIs, Deploying Android Application to the World. Google maps, Using GPS to find current location, Sensors, bluetooth/Wi-Fi Connectivity.

Case Study:

i. List out the points to keep in mind to make you application more attractive. ii. List the controls that make you application attractive.

REFERENCE BOOKS:

1. Reto Meier, “Professional Android 2 Application Development”, Wiley India Pvt Ltd
2. Mark L Murphy, “Beginning Android”, Wiley India Pvt Ltd
3. “Android Application Development All in one for Dummies” by Barry Burd, Edition: I
4. “Android”, Dixit, Prasanna Kumar Vikas Publications, New Delhi 2014, ISBN: 9789325977884
5. Maclean David, Komatineni Satya,Allen Grant , “Pro Android 5”, Apress Publications2015 ISBN: 978-1-4302-4680-0
6.” Android Programming for Beginners” by Hortan, John, Packet Publication, 2015 ISBN: 978-1-78588-326-2
7. Lauren Darcey and Shane Conder, “Android Wireless Application

Development", Pearson Education, 2nd ed. (2011)
ONLINE READING / SUPPORTING MATERIAL:

1. http://www.developer.android.com

2. http://developer.android.com/about/versions/index.html

3. http://developer.android.com/training/basics/firstapp/index.html

4. http://docs.oracle.com/javase/tutorial/index.htm (Available in the form of free downloadable ebooks also).

5. http://developer.android.com/guide/components/activities.html

6. http://developer.android.com/guide/components/fundamentals.html

7. http://developer.android.com/guide/components/intents-filters.html.

8. http://developer.android.com/training/multiscreen/screensizes.html
Syllabus of BCA (Honours) under CBCS 33 9. http://developer.android.com/guide/topics/ui/controls.html 10. http://developer.android.com/guide/topics/ui/declaring-layout.html

11. http://developer.android.com/training/basics/data-storage/databases.html

ST.JOSEPH'S COLLEGE FOR WOMEN (Autonomous), Visakhapatnam
IV SEMESTER B.C.A. (HONORS) Time: 2hrs/Week
BCA-Ma3-4351 MOBILE APPLICATION DEVELOPMENT USING ANDROID

LIST OF EXPERIMENTS:

1. Develop a program to implement frame layout, table layout and relative layout.

2. Develop a program to implement Text View and Edit Text.

3. Develop a program to implement Auto Complete Text View.

4. Develop a program to implement Button, Image Button and Toggle Button.

5. Develop a program to implement login window using above U1 controls.

6. Develop a program to implement Checkbox.

7. Develop a program to implement Radio Button and Radio Group.

8. Develop a program to implement Progress Bar.

9. Develop a program to implement List View, Grid View, Image View and Scroll View.

10. Develop a program to implement Custom Toast Alert.

11. Develop a program to implement Date and Time Picker.

12. Develop a program to create an activity. Develop a program to implement new activity using explicit intent and implicit intent.

13. Develop a program to implement content provider.

14. Develop a program to implement service.

15. Develop a program to implement broadcast receiver.

16. Develop a program to implement sensors.

17. Develop a program to build Camera.

18. Develop a program for providing Bluetooth connectivity.

19. Perform CRUD operations using SQLite.

20. Develop a program for JSON parsing.