B.Sc. PROGRAM (4 years Honors)
Draft proforma for Syllabus framing
2020-21

B.Sc.
Computer Science

Members of BOS(Contact details)		
Dr. P.Venkateshwara Rao	Chairman, Dept. of CSE, ANUR.	9441447037, venkat.aknu@gmail.com
Sri.D.V.S.Suryanarayana	Member, MVNJS & RVR College of A&S, Malkipuram	9666035667, suriyadesineedi@gmail.com
Dr.K.Satyarajesh	Member, HoD Dept. of Computer Science, CSTS Govt Degree College, Jangareddygudem	9494477497, ksatyarajeshcse@gmail.com
Mr.D.Dasu	Coordinator, Dept. of CSE, ANUR	9948677122, dasudasari@gmail.com
Mr.K.Krishna Reddy	Industrialist, Krify Software Technologies(P) LTD, Kakinada	7382211111 kkreddy@krify.net
Mr.M.Suresh	Senior Software Analyst, TCS, Hyderabad, Telangana	9866994478 sureshmallina@gmail.com
Rapaka Ephraim	B.Sc.(CS) Student, MVN JS & RVR College, Malikipuram	9866621025 ephraim143.e1@gmail.com
Tirumala Veera Krishna Sneha	B.Sc.(CS) Student, MVN JS & RVR College, Malikipuram	9705773961 snehatirumalas@gmail.com

TABLE OF CONTENTS

Note: BOS is to providefinal soft copy in PDF and word formats and four copies of hard copies in bounded form to the office of Dean Academic affairs.

1. RESOLUTIONS OF THE BOARD OF STUDIES

Meeting held on: 22.01.2021.Time:10 A.M
At: Adikavi Nannaya University, RJY

Agenda:

1. Adoption of revised-common program structure and revising/updating course - wise syllabi (in the prescribed format) as per the guidelines issued by APSCHE.
2. Adoption of regulations on scheme of examination and marks/grading system of the University UG programs.
3. Preparation of Model question papers in prescribed format.
4. List of equipment/software requirement for each lab/practical
5. Eligibility of student for joining the course
6. Eligibility of faculty for teaching the course
7. List of paper-setters/paper evaluators with phone, email-id in the prescribed format

Members present:

Dr. P.Venkateshwara Rao	Chairman, Dept. of CSE, ANUR.
Sri.D.V.S.Suryanarayana	Member, MVNJS & RVR College of A&S, Malkipuram
Mr.D.Dasu	Coordinator, Dept. of CSE, ANUR

Resolutions:

2. Resolved to adopt the revised-common program structure and revising/updating course-wise syllabi (in the prescribed format) as per the guidelines issued by APSCHE.
3. Resolved to adopt the regulations on scheme of examination and marks/grading system of the University UG programs.
4. Resolved to prepare the Model question papers in prescribed format.
5. Resolved to give the list of equipment/software requirement for each lab/practical
6. Resolved the eligibility of student for joining the course
7. Resolved the eligibility of faculty for teaching the course
8. Resolved to give the list of paper-setters/paper evaluators with phone, email-id in the prescribed format

2. DETAILS OF PAPER TITLES & CREDITS

Sem	Course no.	Course Name	Course type (T/L/P)	Hrs/Week (Arts/ Commerce:5 and Science: 4+2)	Credits (Arts/ Commerce:4 and Science:4+1)	Max. Marks Cont/ Internal/Mid Assessment	Max. Marks Sem-end Exam
I	1	Problem Solving in C	T	4	4	25	75
		Problem Solving in C Lab	L	2	1	-	50
II	2	Data Structures using C	T	4	4	25	75
		Data Structures using C Lab	L	2	1	-	50
III	3	Database Management System	T	4	4	25	75
		Database Management System Lab	L	2	1	-	50
IV	4	Object Oriented Programming using Java	T	4	4	25	75
		Object Oriented Programming using Java Lab	L	2	1	-	50
	5	Operating Systems	T	4	4	25	75
		Operating Systems Lab using C/Java	L	2	1	-	50

Note: *Course type code: T: Theory, L: Lab, P: Problem solving

a. Proposed combination subjects: Computer Applications, Information Technology

b. Student eligibility for joining in the course:

c. Faculty eligibility for teaching the course:

d. List of Proposed Skill enhancement courses with syllabus, if any:

e. Any newly proposed Skill development/Life skill courses with draft syllabus and required resources

f. Required instruments/software/ computers for the course (Lab/Practical course-wise required i.e., for a batch of 15 students)

Sem. No.	Lab/Practical Name	Names of Instruments/Software/ computers required with specifications	Brand Name	Qty Required
1	Problem Solving in C Lab	Intel desktop PC(80GB HDD,512MB DDR), Windows OS, C compiler with supporting editors		15
2	Data Structures using C Lab	Intel desktop PC(80GB HDD,512MB DDR), Windows OS, C compiler with supporting editors		15
3	Database Management System Lab	Intel desktop PC(80GB HDD,512MB DDR), Windows OS, Oracle 8i/9i or SQL Server, MY SQL		15
4	Object Oriented Programming using Java Lab	Intel desktop PC(80GB HDD,512MB DDR), Windows OS, JDK		15
5	Operating Systems Lab using C/Java	Intel desktop PC(80GB HDD,512MB DDR), Windows OS, C compiler with supporting editors, JDK		15

g. List of Suitable levels of positions eligible in the Govt/Pvt organizations

Suitable levels of positions for these graduates either in industry/govt organization like, technical assistants/ scientists/school teachers, clearly define them, with reliable justification

S.No	Position	Company/ Govt organization	Remarks	Additional skills required, if any
1	Software Programmer	IT Industry		
2	Software Developer	IT Industry		
3	Software Engineer	IT Industry		
4	Program Manager	IT Industry		
5	Clerk/PO	Banking Industry		
6	IT Specialist	Banking Industry		
7	Teacher/Lecturer/Asst.Prof	Education Institutes		
8	DB Admin	IT Industry/Medical		

h. List of Govt. organizations / Pvt companies for employment opportunities or internships or projects

S.No	Company/ Govt organization	Position type	Level of Position			
1	Software Development Industry					
2	E-Commerce Industry					
3	Medical Industry					
4	IT Industry					
5	Banking Industry					
6	Education Industry					

i. Any specific instructions to the teacher /paper setters/Exam-Chief Superintendent

3. PROGRAM OBJECTIVES, OUTCOMES, CO-CURRICULAR AND ASSESSMENT METHODS

B.Sc.	Computer Science

1. Aim and objectives of UG program in Subject: Computer Science

The Objectives of this Program describes what students are expected to know and be able to do by the time of graduation. The Computer Science Department's Bachelor of Science program must enable students to attain, by the time of graduation:

- An ability to identify, formulate and develop solutions to computational challenges.
- An ability to design, implement and evaluate a computational system to meet desired needs within realistic constraints.
- An ability to function effectively on teams to accomplish shared computing design, evaluation, or implementation goals.
- An understanding of professional, ethical, legal, security, and social issues and responsibilities for the computing profession.
- An ability to communicate and engage effectively with diverse stakeholders.

- An ability to analyze impacts of computing on individuals, organizations, and society.
- Recognition of the need for and ability to engage in continuing professional development.
- An ability to use appropriate techniques, skills, and tools necessary for computing practice.
- Effectively utilizing their knowledge of computing principles and mathematical theory to develop sustainable solutions to current and future computing problems.
- Developing and implementing solution based systems and/or processes that address issues and/or improve existing systems within in a computing based industry.

2. Learning outcomes of Subject Computer Science:

- Students will be able to communicate in written and oral forms in such a way as to demonstrate their ability to present information clearly, logically, and critically.
- Students will be able to apply mathematical and computing theoretical concepts in solution of common computing applications, such as computing the order of an algorithm.
- Students will be able to complete successfully be able to program small-to-mid-size programs on their own. Sufficient programming skills will require use of good practice, e.g., good variable names, good use of computational units, appropriate commenting strategies.
- Students will be able to use appropriately system design notations and apply system design engineering process in order to design, plan, and implement software systems
- In a self-selected area of depth in Computing, students will demonstrate a depth of knowledge appropriate to graduate study and/or lifelong learning in that area. Students should be able to read for understanding materials in that area beyond those assigned in coursework.
- Students will be prepared for a career in an information technology oriented business or industry, or for graduate study in computer science or other scientific or technical fields.
- Use systems development, word-processing, spreadsheet, and presentation software to solve basic information systems problems

3. Recommended Skill enhancement courses: (Titles of the courses given below and details of the syllabus for 4 credits (i.e., 2 units for theory and Lab/Practical) for 5 hrs class-cum-lab work.

4. Recommended Co-curricular activities: (Co-curricular Activities should not promote copying from text book or from others' work and shall encourage self/independent and group learning)

A. Measurable:
   1. Assignments
   2. Student seminars (Individual presentation of papers)
   3. Quiz Programmes
   4. Individual Field Studies/projects
   5. Group discussion
   6. Group/Team Projects

B General
   1. Collection of news reports and maintaining a record of paper-cuttings relating to topics covered in syllabus
   2. Group Discussions
   3. Watching TV discussions and preparing summary points recording personal observations etc., under guidance from the Lecturers
   4. Any similar activities with imaginative thinking.

5. Recommended Continuous Assessment methods:

Some of the following suggested assessment methodologies could be adopted;

- The oral and written examinations (Scheduled and surprise tests),
- Closed-book and open-book tests,
- Coding exercises,
- Practical assignments and laboratory reports,
- Observation of practical skills,
- Individual and group project reports,
- Efficient delivery using seminar presentations,
- Viva voce interviews.
- Computerized adaptive testing, literature surveys and evaluations,
- Peers and self-assessment, outputs form individual and collaborative work

4. DETAILS OF COURSE-WISE SYLLABUS

BSc	Computer Science (Semester: I)	Credits: 4
Paper: 1	PROBLEM SOLVING IN C	Hrs/Wk: 4

1. Aim and objectives of Course:

This course aims to provide exposure to problem-solving through programming. It introduces the concepts of the C Programming language.

2. Learning outcomes of Course:

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

1. Understand the evolution and functionality of a Digital Computer.
2. Apply logical skills to analyse a given problem
3. Develop an algorithm for solving a given problem.
4. Understand 'C' language constructs like Iterative statements, Array processing, Pointers, etc.
5. Apply 'C' language constructs to the algorithms to write a 'C' language program.

3. Detailed Syllabus: (Five units with each unit having 12 hours of class work)

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.

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.

REFERENCES:

1. Ashok N Kamthane: Programming with ANSI and Turbo C, Pearson Edition Publications.
2. YashavantKanetkar - Let Us ‘C’ – BPB Publications.

4. Details of Lab Syllabus: **Problem solving in C LAB**

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 refection 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 two N 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

5. RECOMMENDED CO-CURRICULAR ACTIVITIES:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A. Measurable

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity))
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity

B. General

1. Group Discussion
2. Try to solve MCQ's available online.
3. Others

6. RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Problem-solving exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports like "Creating Text Editor in C".
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

5. MODEL QUESTION PAPER (Sem-end. Exam)

BSc	Computer Science(Semester: I)	Max. Marks: 75M
Paper: 1	PROBLEM SOLVING IN C	3Hrs

Answer any 5 question **5X5 = 25M**

1. Explain Block diagram of Computer.
2. Define an Algorithm. What are the key features of an algorithm?
3. Write about go to statement with syntax and example.
4. Dynamic memory allocation.
5. Explain pointers in arrays.
6. How to write data from files with example?
7. Write about enumerated data types.
8. Briefly explain various types of recursions.

Answer following question **5X10 = 50M**

9. a) Briefly explain about generations of computers.

(OR)

b) What is a Flowchart? Explain significance with an example.

10. a) Explain basic data types in C?

(OR)

b) Explain about iterative statements available in C.

11. a) What is an Array? Explain different types of arrays with examples.

(OR)

b) What is a string? Explain various string handling functions available in C.

12. a) Define a function. Explain the passing parameter mechanism.

(OR)

b) Explain about Structure with syntax and example in detail.

13. a) Define and use of a pointer and write a ‘C’ program on swapping of two numbers using pointers.

(OR)

b) Explain file modes in detail.

BSc	Computer Science (Semester: II)	Credits: 4
Paper: 2	DATA STRUCTURES USING C	Hrs/Wk: 4

1. Aim and objectives of Course:

To introduce the fundamental concept of data structures and to emphasize the importance of various data structures in developing and implementing efficient algorithms.

2. Learning outcomes of Course:

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

1. Understand available Data Structures for data storage and processing.
2. ComprehendData Structure and their real-time applications - Stack, Queue, Linked List, Trees and Graph
3. Choose a suitable Data Structures for an application
4. Develop ability to implement different Sorting and Search methods
5. Have knowledge onData Structures basic operations like insert, delete, search,update and traversal
6. Design and develop programs using various data structures
7. Implement the applications of algorithms for sorting, pattern matching etc

3. Detailed Syllabus: (Five units with each unit having 12 hours of class work)

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.

Text Books:

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

References:

1. “Data Structures Using C” Balagurusamy E. TMH

4. Details of Lab Syllabus: **Data Structures Using C Lab**

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 begging of an array
    b. Insert an element at given index of array
    c. Update a element using a values 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 Liked List.
5. Write Programs to implement the Queue operations using an array.
6. Write Programs to implement the Queue operations using Liked 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 Dijkstrar'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

5. RECOMMENDED CO-CURRICULAR ACTIVITIES:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A. Measurable

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity))
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity

B. General

1. Group Discussion
2. Try to solve MCQ's available online.
3. Others

6. RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Problem-solving exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports like "Creating Text Editor in C".
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

MODEL QUESTION PAPER (Sem-end. Exam)

BSc	Computer Science(Semester: II)	Max. Marks: 75M
Paper: 2	DATA STRUCTURES USING C	3Hrs

Answer any 5 question **5X5 = 25M**

1. Explain about Abstract Data Types.
2. Define linear and non-linear data structures.
3. Explain Atomic Linked List.
4. What are the applications of stacks?
5. What is priority queue?
6. Explain about binary search tree.
7. Define sorting. What are the advantages and disadvantages of merge sort?
8. Briefly explain various representations of Graphics.

Answer following question **5X10 = 50M**

9. a) What are primitive and non-primitive data structures with an example?

(OR)

b) Explain different approaches to designing an algorithm.

10. a) Explain different types of arrays.

(OR)

b) What is linked list? Explain different types of linked lists in data structures.

11. a) What is stack? Write ADT. Explain various operations of stack.

(OR)

b) What is a Deque? What are the different techniques used to represent Deque? Explain.

12. a) Write about different tree traveling techniques and write an algorithm for traveling techniques.

(OR)

b) Explain different applications and properties of binary tree.

13. a) Write about various Graph Travelling techniques.

(OR)

b) What is searching? Explain Linear Search Algorithm with example.

BSc	Computer Science (Semester: III)	Credits: 4
Paper: 3	DATABASE MANAGEMENT SYSTEMS	Hrs/Wk: 4

1. Aim and objectives of Course:

The objective of the course is to introduce the design and development of databases with special emphasis on relational databases.

2. Learning outcomes of Course:

Upon successful completion of the course, a student will be able to:
On completing the subject, students will be able to:

1. Gain knowledge of Database and DBMS.
2. Understand the fundamental concepts of DBMS with special emphasis on relational data model.
3. Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database
4. Model databaseusing ER Diagrams and design database schemas based on the model.
5. Create a small database using SQL.
6. Store, Retrieve data in database.

3. Detailed Syllabus: (Five units with each unit having 12 hours of class work)

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 3rd 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.

TEXT BOOKS:

1. Database System Concepts by Abraham Silberschatz, Henry Korth, and S. Sudarshan, McGrawhill
2. Database Management Systems by Raghu Ramakrishnan, McGrawhill

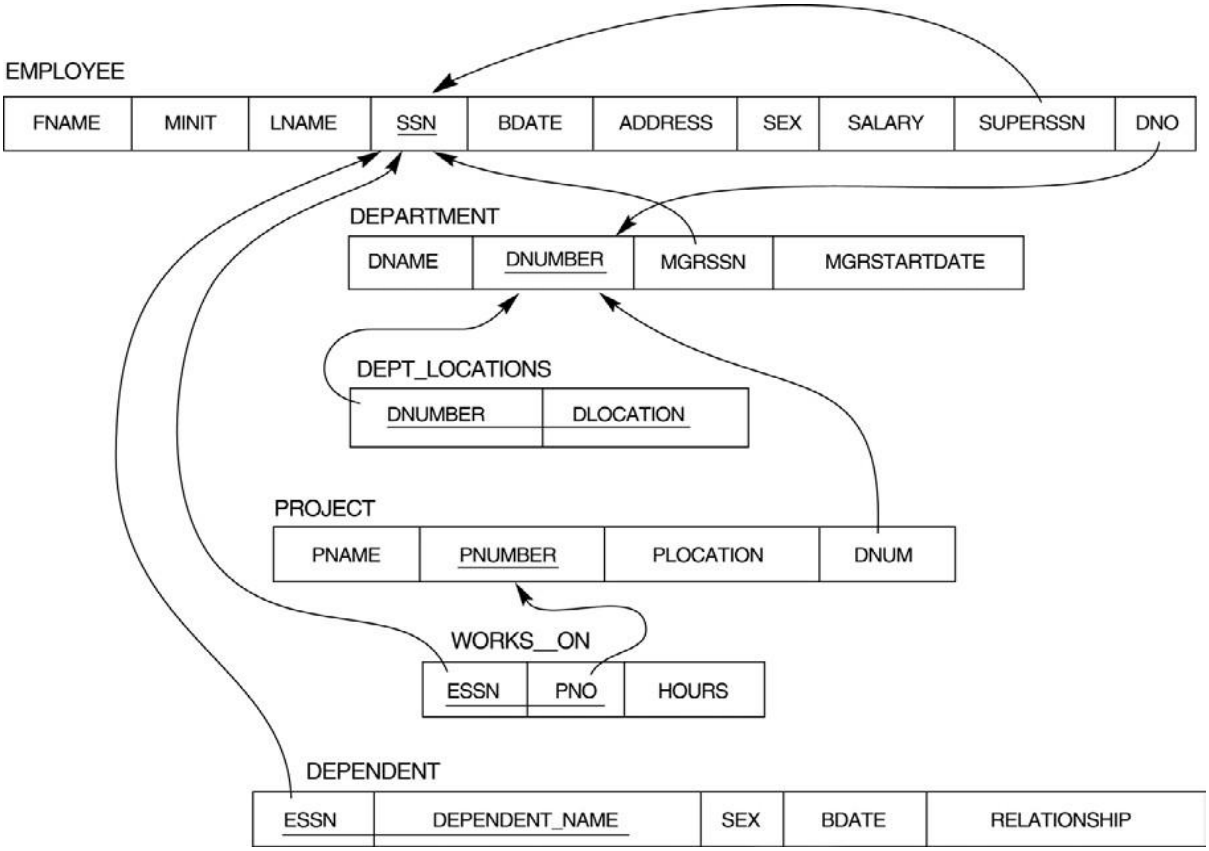
REFERENCES:

1. Principles of Database Systems by J. D. Ullman
2. Fundamentals of Database Systems by R. Elmasri and S. Navathe
3. SQL: The Ultimate Beginners Guide by Steve Tale.

4. Details of Lab Syllabus: **DATABASE MANAGEMENT SYSTEMS LAB**

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 above 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.

5. RECOMMENDED CO-CURRICULAR ACTIVITIES:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A. Measurable

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity))
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity

B. General

1. Group Discussion
2. Try to solve MCQ's available online.
3. Others

6. RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Problem-solving exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports like "Creating Text Editor in C".
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

MODEL QUESTION PAPER (Sem-end. Exam)

BSc	Computer Science(Semester: III)	Max. Marks: 75M
Paper: 3	DATABASE MANAGEMENT SYSTEMS	3Hrs

Answer any 5 question **5X5 = 25M**

1. Explain disadvantages of file processing system?
2. Explain the concept of entity and entity set with suitable example.
3. Explain about various attribute classification.
4. What are the advantages of Relational algebra? Explain.
5. Explain various types of keys.
6. Explain the selection command with an example.
7. Explain sub queries.
8. Explain structure of PL/SQL.

Answer following question **5X10 = 50M**

9. a) With a neat diagram, explain the architecture of a DBMS.

(OR)

b) Explain about Data Models.

10. a) Explain about Specialization and Generalization in EER model.

(OR)

b) What is ER-Modeling? Write advantages and disadvantages of ER-Modelling.

11. a) What is Functional Dependency? Explain difference between 3NF and BCNF?

(OR)

b) What is relational model? Write about key features of relational model.

12. a) What is SQL? Explain different types of commands in SQL.

(OR)

b) What is Nested Queries? How to create them? Discuss it with relevant example.

13. a) Explain steps in creating a PL/SQL Program.

(OR)

b) Explain about Triggers and types of triggers.

BSc	Computer Science (Semester: IV)	Credits: 4
Paper: 4	OBJECT ORIENTATED PROGRAMMING USING JAVA	Hrs/Wk: 4

1. Aim and objectives of Course:

To introduce the fundamental concepts of Object-Oriented programming and to design & implement object oriented programming concepts in Java.

2. Learning outcomes of Course:

1. Understand the benefits of a well-structured program
2. Understand different computer programming paradigms
3. Understand underlying principles of Object-Oriented Programming in Java
4. Develop problem-solving and programming skills using OOP concepts
5. Develop the ability to solve real-world problems through software development in high-level programming language like Java

3. Detailed Syllabus: (Five units with each unit having 12 hours of class work)

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 uingFileInputStream, 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

TEXT BOOKS:

1. Core Java: An Integrated Approach, Authored by Dr. R. Nageswara Rao &Kogent Learning Solutions Inc.

2. E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw- Hill Company.

REFERENCES:

1. John R. Hubbard, Programming with Java, Second Edition, Schaum's outline Series, TMH.

2. Deitel&Deitel. Java TM: How to Program, PHI (2007)

4. Details of Lab Syllabus: **Object Orientated Programming using Java Lab**

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 works 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

5. RECOMMENDED CO-CURRICULAR ACTIVITIES:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A. Measurable

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity))
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity

B. General

1. Group Discussion
2. Try to solve MCQ's available online.
3. Others

6. RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Problem-solving exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports like "Creating Text Editor in C".
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

MODEL QUESTION PAPER (Sem-end. Exam)

BSc	Computer Science(Semester: IV)	Max. Marks: 75M
Paper: 4	OBJECT ORIENTED PROGRAMMING USING JAVA	3Hrs

Answer any 5 question **5X5 = 25M**

1. Explain about JVM.
2. Explain about factory methods.
3. Explain about 'this' keyword with example.
4. Explain about Type casting.
5. Define Abstract class and Abstract method.
6. Explain Zipping and Unzipping files.
7. How to terminate a thread.
8. Explain JDBC.

Answer following question **5X10 = 50M**

9. a) Explain Looping statements in JAVA.

(OR)

b) Explain operators and types of operators.

10. a) Explain Inheritance and types of Inheritance.

(OR)

b) Explain constructors and types of constructors with an example.

11. a) Describe Interface? Critically explain and define Accessing Interface variable.

(OR)

b) Explain concept of Exception handling.

12. a) Explain the concept of Creating a file using File Writer using an example program.

(OR)

b) Discuss Thread Life Cycle.

13. a) Define Applet. Explain how to create an Applet.

(OR)

b) Explain the procedure to connect Oracle Database using jdbc-odbc driver.

BSc	Computer Science (Semester: IV)	Credits: 4
Paper: 5	OPERATING SYSTEMS	Hrs/Wk: 4

1. Aim and objectives of Course:

This course aims to introduce the structure and organization of a file system. It emphasizes various functions of an operating system like memory management, process management, device management, etc.

2. Learning outcomes of Course:

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

1. Know Computer system resources and the role of operating system in resource management with algorithms
2. Understand Operating System Architectural design and its services.
3. Gain knowledge of various types of operating systems including Unix and Android.
4. Understand various process management concepts including scheduling, synchronization, and deadlocks.
5. Have a basic knowledge about multithreading.
6. Comprehenddifferent approaches for memory management.
7. Understand and identify potential threats to operating systems and the security features design to guard against them.
8. Specify objectives of modern operating systems and describe how operating systems have evolved over time.
9. Describe the functions of a contemporary operating system

3. Detailed Syllabus: (Five units with each unit having 12 hours of class work)

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, ProcessAbstraction, ProcessHierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Preemptive and Preemptive Scheduling Algorithms.

UNIT III

Process Management: Deadlock, Deadlock Characterization, Necessary andSufficient Conditions forDeadlock, 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; MemoryAllocation 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.

TEXT BOOKS:

1. Operating System Principles by Abraham Silberschatz, Peter Baer Galvin and Greg Gagne (7^{th}Edition) Wiley India Edition.
2. Operating Systems: Internals and Design Principles by Stallings (Pearson)

REFERENCES:

1. Operating Systems by J. Archer Harris (Author), Jyoti Singh (Author) (TMH)
2. Online Resources for UNIT V

4. Details of Lab Syllabus: **Operating Systems Lab using C/Java**

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
12. Write a program for Bankers Algorithm for Dead Lock Avoidance
13. Implement Bankers Algorithm Dead Lock Prevention.
14. Simulate all Page replacement algorithms.
    a) FIFO
    b) LRU
    c) LFU
15. Simulate Paging Techniques of memory management

5. RECOMMENDED CO-CURRICULAR ACTIVITIES:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A. Measurable

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity))
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity

B. General

1. Group Discussion
2. Try to solve MCQ's available online.
3. Others

6. RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

2. The oral and written examinations (Scheduled and surprise tests),
3. Closed-book and open-book tests,
4. Problem-solving exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports like "Creating Text Editor in C".
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

MODEL QUESTION PAPER (Sem-end. Exam)

BSc	Computer Science(Semester: IV)	Max. Marks: 75M
Paper: 5	OPERATING SYSTEMS	3Hrs

Answer any 5 question **5X5 = 25M**

1. Write about Resource Abstraction.
2. Write about the process and the process state.
3. Explain threading issues.
4. Explain about process Synchronization.
5. Discuss some necessary and sufficient conditions for deadlock.
6. Explain about Virtual memory.
7. Explain about shared memory.
8. Write about file types.

Answer following question **5X10 = 50M**

9. a) Explain various types of Operating Systems.

(OR)

b) What is Operating System? Explain functions of Operating System.

10. a) Explain in detail about Process Scheduling.

(OR)

b) Explain system view of the process and resources.

11. a) Explain about deadlock Detection and recovery.

(OR)

b) Discuss classical process synchronization problems.

12. a) Explain the following

i) Segmentation

ii) Fixed and variable partitions.

(OR)

b) Explain in detail about Demand-paging.

13. a) Explain Authentication and Internal Access Authorization.

(OR)

b) Explain Android Development Framework.