



K.L.E. Society's

**BACHELOR OF COMPUTER  
APPLICATIONS RLS INSTITUTE, BELAGAVI**

**2024-25**

**FIRST**

**YEAR**

**SYLLABUS**

**FIRST**

**SEMESTER**

**I SEMESTER COURSE STRUCTURE**

SEMESTER-I						
Category	Subject Code	Title of the Paper	Teaching hours/week	Examination	Duration of	Credit

				Practic		Theor y/ Practi cal	Tota l	exams (Hrs)	
AECC	24CA101	English	3	0	20	80	100	3	3
AECC	24CA102	Kannada	3	0	20	80	100	3	3
	24CA103	Hindi							
DSC1	24CA104	Programming in C	4	0	20	80	100	3	4
DSC2	24CA105	Programming in C Lab	0	4	10	40	50	2	2
DSC3	24CA106	Web Programming	4	0	40	80	100	3	4
DSC4	24CA107	Web Programmi ng Lab	0	4	10	40	50	2	2
	24CA108	Mathematics	4	0	20	80	100	3	4
	24CA109	Constitutio nal Values	2	0	10	40	50	2	2
<b>Total Credits/Marks</b>			<b>20</b>	<b>8</b>			<b>650</b>		<b>24</b>

<b>Year</b>	<b>I</b>	<b>Course Code: 24CA104</b>	<b>Credits</b>	<b>4</b>
<b>Semester</b>	<b>I</b>	<b>Course Title: Programming in C</b>	<b>Hours</b>	<b>52</b>
<b>Formative Assessment Marks:</b>		<b>Summative Assessment Marks:</b>		<b>Duration of ESA: 03 hrs.</b>

Course Outcomes	At the end of the course, the students should be able to: o Understand classification of computers, its features and parts, software and its types o Apply techniques of problem solving to design C code o Read, understand and trace the execution of programs written in C language o Apply programming control structures for a given problem to create C code o Understand derived datatypes and develop C code using arrays/ strings o Understand user defined functions and datatypes to develop C code	
Unit No	Course Content	Hours
UNIT - 1	<p><b>Introduction to programming overview of C:</b> Basic Programming concept: Algorithms flowchart, Modular Programming and structured programming. Introduction, Importance of 'C', Sample 'C' Programs, Basic structure of 'C' programs, Programming style, Executing a 'C' Program. 'C' Tokens, keywords, and identifiers, constants, variables, data types, declaration of variables, assigning values to variables, defining symbolic constants.</p> <p><b>Input and Output statements:</b> Input and Output statements, reading a character, writing characters, formatted input, formatted output statements.</p> <p><b>Operators, Expression:</b> Arithmetic operators, Relational operators. Logical operators, Assignment operators, increment and decrement operators, conditional operators, bitwise operators, special operators, some computational problems, type conversion in expressions, operator precedence and associativity. Mathematical functions.</p>	14
UNIT - 2	<p><b>Control structures &amp; Array:</b></p> <p><b>Decision making and branching:</b> simple IF statement, IF-ELSE statement, nesting of IF ELSE statements, ELSE -IF ladder, switch statement, ? operator, GOTO statement</p> <p><b>Decision making and looping:</b> FOR, WHILE , Do-While and jumps in loops</p> <p><b>Arrays:</b></p> <p>One dimensional array, Two- dimensional arrays, initializing two dimensional array, Multidimensional arrays.</p>	12

<b>UNIT - 3</b>	<p><b>Strings, User defined functions and Structure:</b></p> <p>Declaring and initializing string variables, reading string from terminal, writing string to screen, arithmetic operations on characters, putting strings together. Comparison of two strings, string handling functions-strlen, strcat, strcmp, strcpy.</p> <p>Need for user-defined functions, a multi-functional program, the form of 'C' function, return values and their types, calling a function, category of functions, recursion, functions with arrays.</p> <p>Structure definition, giving values to members, structure initialization, comparison of structure variables, array as structure, array within structure, union.</p>	14
<b>UNIT - 4</b>	<p><b>Pointers and file operations:</b> Understanding pointers, accessing the address of variables, declaring and initializing pointers, accessing a variable through its pointer definition.</p> <p>Basic file operations: Naming a file, opening a file, reading data from file, writing data to a file, and closing a file. Defining, Opening and closing a file. Input / Output operations on files: getc, putc, getw, putw, fprintf, fscanf.</p>	12

**Text Books:**

1. Fundamentals of Computers, E. Balaguruswamy (McGraw Hill)
2. Anil V. Choudhuri, The Art of Programming through Flowchart and Algorithms, Laxmi Pub.
3. E. Balaguruswamy: Programming in ANSI C (TMH)
4. Kamthane: Programming with ANSI and TURBO C (Pearson Education)
5. V. Rajaraman: Programming in C (PHI –EEE)
6. P.B. Kottur: Programming in C (Sapna Book House)

**References:**

1. P. K. Sinha & Priti Sinha: Computer Fundamentals (BPB)
2. C: The Complete Reference, By Herbert Schildt.
3. Kernighan & Ritchie: The C Programming Language (PHI)
4. S. Byron Gottfried: Programming with C (TMH)
5. Yashwant Kanitkar: Let us C

<b>Year</b>	<b>I</b>	<b>Course Code: 24CA108</b>	<b>Credits</b>	<b>4</b>
-------------	----------	-----------------------------	----------------	----------

<b>Semester</b>	<b>I</b>	<b>Course Title: Mathematics</b>	<b>Hours</b>	<b>52</b>
<b>Formative Assessment Marks:</b>	<b>Summative Assessment Marks:</b>		<b>Duration of ESA: 03 hrs.</b>	
<b>Course Outcomes</b>	<p>At the end of the course, the students should be able to:</p> <ul style="list-style-type: none"> <li>● Recognize when set theory is applicable to real-life situations, solve real life problems, and communicate real-life problems and solutions to others.</li> <li>● Define determinants and understand their relation to matrices. ● Demonstrate a clear understanding of fundamental concepts. ● Apply problem-solving techniques to solve recruitment-based problems. ● Use appropriate strategies and shortcuts to improve speed and accuracy in solving aptitude problems during recruitment processes.</li> </ul>			
<b>Unit No</b>	<b>Course Content</b>			<b>Hours</b>
<b>UNIT – 1</b>	<b>Sets:</b> Introduction, Definition and Representation, Types of sets, Operations on sets, Venn Diagram, Complement of a set, Problems on Union and Intersection of two sets, De- Morgan’s Law.			14
<b>UNIT – 2</b>	<b>Linear Algebra:</b> Definition of Matrices. Types of matrices. Operations on matrices. Transpose, Co-factor, Adjoint and Inverse of a matrix. Definition and properties of Determinants. Solving pair of linear equations.			12
<b>UNIT – 3</b>	<b>Fundamentals of Logic:</b> Introduction, Propositions, Logical Connectives, Compound Propositions, Truth Tables, Tautology, Contradiction and Contingency, Logical Equivalence.			12
<b>UNIT – 4</b>	<p><b>Relations, Functions and Graph Theory:</b> Definition relation, Types of Relations, Definition of Function, Types of Function, Composition of Functions and Invertible Functions.</p> <p>Graph Theory: Introduction to graphs, Graph terminology, Representing graph and graph isomorphism, Connectivity, Eulerian and Hamilton graph, Planar graph.</p>			14

**Text Books:**

1. NCERT 11<sup>th</sup> and 12<sup>th</sup> Textbooks

**References:**

1. Quantitative Aptitude by Dr. R. S. Aggarwal
2. Aptitude by S. Chand

<b>Year</b>	<b>I</b>	<b>Course Code: 24CA106</b>	<b>Credits</b>	<b>4</b>
<b>Semester</b>	<b>I</b>	<b>Course Title: Web Programming</b>	<b>Hours</b>	<b>52</b>
<b>Formative Assessment Marks:</b>	<b>Summative Assessment Marks:</b>		<b>Duration of ESA: 03 hrs.</b>	
<b>Course Outcomes</b>	At the end of the course, the students should be able to: <ul style="list-style-type: none"><li>o Understand classification of computers, its features and parts, software and its types</li><li>o Apply techniques of problem solving to design C code</li><li>o Read, understand and trace the execution of programs written in C language</li><li>o Apply programming control structures for a given problem to create C code</li><li>o Understand derived datatypes and develop C code using arrays/ strings</li><li>o Understand user defined functions and datatypes to develop C code</li></ul>			
<b>Unit No</b>	<b>Course Content</b>			<b>Hours</b>
<b>UNIT – 1</b>	<b>HTML &amp; HTML5</b> - History of Internet, world wide web, Introduction and Role of Web Technology in today 's Scenario and Internet Protocols related to web. Introduction to HTML, Basic structure of HTML document, Differences between HTML and HTML5, Text Formatting Tags, Lists, Image, Tables, Links, Form Elements, Multimedia tags, Frames, image maps.			13
<b>UNIT – 2</b>	<b>Dynamic HTML</b> -Event model: introduction, event ON CLICK, event ON LOAD – error handling with ON ERROR, tracking the mouse with event, more DHTML events. Filters and Transitions. Cascading Style Sheet (CSS3), Types of Style Sheets (In-line, External and Embedded), CSS Selectors.			13

<b>UNIT – 3</b>	<p><b>Java Script</b>-Role of Scripting Languages in Web Development, History of JavaScript, Data types, Operators, Control and Looping Structure.</p> <p><b>Arrays and Strings</b>-Arrays, User Defined and Inbuilt Functions, String, Character and Date Functions, Document Object Model, Object Hierarchy in DOM, Event Handling, and Writing Client-Side Validation for HTML Form Elements.</p>	14
<b>UNIT – 4</b>	<p><b>Introduction to XML</b>-Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS; XSLT style sheets; XML processors; Web services.</p>	12

**Text Books:**

1. Robert W. Sebestra, —Programming the World Wide Web, 7th Edition Addison Wesley Publication, 2013.
2. A beginner's guide to HTML, CSS, Javascript, and Web Graphics, by Jennifer Niederst Robbins.
3. Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 by Robin Nixon

**References:**

1. Chris Bates, —Web Programming: Building Internet Applications, 3rd Edition Wiley 2009.
2. PHP Objects, Patterns, and Practice by Matt Zandstra

<b>Year</b>	<b>I</b>	<b>Course Code: 24CA105</b>	<b>Credits</b>	<b>2</b>
<b>Semester</b>	<b>I</b>	<b>Course Title: Programming in C Lab</b>	<b>Hours</b>	<b>30</b>
<b>Formative Assessment Marks:</b>		<b>Summative Assessment Marks:</b>		<b>Duration of ESA: 02 hrs.</b>
Course Outcomes		<p>At the end of the course, the students should be able to:</p> <ul style="list-style-type: none"> <li>· Develop a C program.</li> <li>· Control the sequence of the program and give logical outputs.</li> <li>· Implement strings in your C program.</li> <li>· Store different data types in the same memory.</li> <li>· Manage I/O operations in your C program.</li> </ul>		

	· Repeat the sequence of instructions and points for a memory location.
--	---

1. Write a C Program to find largest of three numbers.
2. Write a C Program to find simple and compound interest.
3. Write a C Program to find whether a given number is prime number or not.
4. Write a C Program to generate and print first 'n' FIBONACCI numbers.
5. Write a C Program that reverse a given integer number and check whether the number is palindrome or not.
6. Write a C Program to swap two numbers without using third variable.
7. Write a C Program to find if a character is alphabetic or numeric or special character.
8. C Program to Check Whether a Number is Positive, Negative, or Zero
9. Write a C Program to display result of a student using switch statement.
10. Write a C Program to print pyramid patterns using loops.
11. C Program to Generate Multiplication Table
12. C Program to Find the Maximum and Minimum in an Array
13. Write a C Program to read two matrices and perform addition and subtractions of two matrices.
14. Write a Program to find the factorial of a number using function.
15. Write a C Program to compute the sum of even numbers and the sum of odd numbers using a function.
16. Write a C Program to accept a sentence and convert all lowercase characters to uppercase and vice -versa.
17. Write a C Program to find the length of a string without using the built – in function.
18. Write a C Program using structures, to accept different goods with the number, price and date of purchase and display those.
19. C Program to Store Student Records as Structures and Sort them by Name.
20. Write a C Program to demonstrate pointers in C.

<b>Year</b>	<b>I</b>	<b>Course Code: 24CA107</b>	<b>Credits</b>	<b>2</b>
<b>Semester</b>	<b>I</b>	<b>Course Title: Web Programming Lab</b>	<b>Hours</b>	<b>30</b>
<b>Course Pre requisite, if any</b>		NA		



<b>Formative Assessment Marks:</b>	<b>Summative Assessment Marks:</b>	<b>Duration of ESA: 02 hrs.</b>
Course Outcomes	<p>At the end of the course, the students should be able to:</p> <ul style="list-style-type: none"> <li>· Design and develop web applications.</li> <li>· Explain client and server-side scripting and their applicability.</li> <li>· Create scripts using JavaScript in a web page.</li> <li>· Integrate JavaScript in a web page.</li> <li>· Design forms and check for data accuracy.</li> </ul>	

1. Write HTML code to Illustrate text formatting tags.
2. Write HTML code to demonstrate ordered list and unordered list.
3. Write HTML code to demonstrate image tag <img> Tag
4. Write HTML code to demonstrate table tag and its attributes.
5. Write HTML code to demonstrate concept of links
6. Write an HTML code to demonstrate form tag
7. Write an HTML code to demonstrate frame tag
8. Write an HTML program to demonstrate Image maps.
9. Write an HTML program to demonstrate On click event
10. Write a program to demonstrate Onmouseover and Onmouseout event
11. Write a program to demonstrate Keyboard events.
12. Write an HTML program to demonstrate Filters.
13. Write an HTML program to demonstrate Transition
14. Write an HTML code to illustrate external CSS
15. Write a JS program to demonstrate arithmetic operators using form
16. Write a JS program to demonstrate switch statement
17. Write a JS program to demonstrate arrays operations
18. To write a program for get the name of the user from a form and show greeting text
19. Write an XML Program to display Student Details.
20. Write an XML Program to display Employee Details.

# **SECOND**

# **SEMESTE**

# R

## II SEMESTER SUBJECT LIST

<b>SEMESTER-II</b>									
<b>Catego ry</b>	<b>Course Code</b>	<b>Title of the Paper</b>	<b>Teaching hours/week</b>		<b>Examination</b>			<b>Durati onof exams (Hrs)</b>	<b>Cred it</b>
			Lect	Practical IA	Theory	Practical	Total		
AECC	24CA201	English	3	0	20	80	100	3	3
AECC	24CA202	Kannada	3	0	20	80	100	3	3
	24CA203	Hindi							
DSC6	24CA204	Data Structures & Algorithms using C	4	0	20	80	100	3	4
DSC7	24CA205	Data Structures & Algorithms using C Lab	0	4	10	40	50	2	2
DSC8	24CA206	Database Management Systems	4	0	40	80	100	3	4
DSC9	24CA207	Database Management Systems Lab	0	4	10	40	50	2	2
DSC10	24CA208	Operating System	4	0	20	80	100	3	4
SEC2	24CA209	Constitutional Values	2		10	40	50	2	2
<b>Total Hours/ Marks/Credits</b>			<b>20</b>	<b>8</b>			<b>650</b>		<b>24</b>

<b>Year</b>	I	<b>Course code:</b> 24CA204	<b>Credits</b>	4
<b>Semester</b>	II	<b>Course title:</b> Data structures and algorithms using C++	<b>Hours</b>	50
<b>Course Prerequisite, if any</b>	Knowledge of C programming			
<b>Formative assessment marks :</b>	<b>Summative assessment marks :</b>		<b>Duration of ESA : 03 hours</b>	
<b>Course outcomes</b>	<p>At the end of the course the student should be able to:</p> <ul style="list-style-type: none"> <li>· To understand basic data structures, their implementation and some of their standard applications.</li> <li>· To develop the ability to design and analyse basic algorithms and prove their correctness using the appropriate data structure learned in the course.</li> </ul>			
<b>Unit No</b>	<b>Course content</b>			<b>Hours</b>
<b>UNIT – 1</b>	<p><b>Introduction to Data Structures:</b> Definition, Classification of data structures: primitive and non-primitive. Operations on data structures.</p> <p><b>Role of Algorithms in Computing:</b> Algorithms, Algorithms as a technology, Analysing algorithms, Growth of Functions- Asymptotic notations, calculating time complexity, calculating space complexity.</p> <p><b>Searching techniques:</b> Linear search and Binary search <b>Sorting techniques:</b> Bubble sort, Selection sort, Quick sort and Merge sort</p>			<b>12</b>
<b>UNIT – 2</b>	<p><b>Dynamic memory allocation:</b> Meaning of static &amp; dynamic memory allocation; memory allocation and de-allocation functions: malloc, calloc, realloc and free.</p> <p><b>Stack:</b> Definition, Array representation of stack, Operations on stack: Infix, prefix and postfix notations, Conversion of an arithmetic expression from Infix to postfix, Applications of stacks</p> <p><b>Queue:</b> Definition, Array representation of queue, types of queue: Simple queue, circular queue, double ended queue priority queue, operations on all types of queues, applications of queue.</p>			<b>12</b>

<b>UNIT – 3</b>	<p><b>Linked list:</b> Definition, components of linked list, Representation of linked list, Advantages and disadvantages of linked list, Types of linked list: singly linked list, doubly linked list and Circular list, operations linked list: Creation, insertion, deletion, search and display.</p> <p><b>Trees:</b> Tree terminology, Binary tree, Complete Binary Tree, Binary search tree, Tree Traversals, Creation of Binary Tree from traversal methods, Binary Search Tree –Insertion &amp; deletion in BST.</p> <p><b>Graphs:</b> Graph terminology, Representation of graphs, Path matrix, Graph Traversal – BFS (breadth first search) –DFS (depth first search).</p>	<b>12</b>
<b>Unit – 4</b>	<p><b>Greedy Method</b> General method, solving Job sequencing with deadlines Problems. <b>Minimum cost spanning trees:</b> Prim’s Algorithm, Kruskal’s Algorithm with performance analysis. <b>Single source shortest paths:</b>Dijkstra's Algorithm</p> <p><b>Dynamic Programming UV</b> General method with Examples, Multistage Graphs. <b>Transitive Closure:</b>Warshall’s Algorithm. <b>All Pairs Shortest Paths:</b> Floyd's Algorithm, Knapsack problem, Travelling Salesman problem.</p>	<b>14</b>

**Text Books:**

1. Ellis Horowitz and SartajSahni: Fundamentals of Data Structures
2. Tanenbaum: Data structures using C (Pearson Education)

**References:**

1. Y. Kanitkar: Data Structures Using C(BPB)
2. Kottur: Data Structure Using C
3. Padma Reddy: Data Structure Using C

<b>Year</b>	<b>I</b>	<b>Course Code:</b> 24CA206	<b>Credits</b>	<b>4</b>
<b>Semester</b>	<b>II</b>	<b>Course Title:</b> Data Base Management System	<b>Hours</b>	<b>50</b>
<b>Formative Assessment Marks:</b>		<b>Summative Assessment Marks:</b>	<b>Duration of ESA: 03 hrs.</b>	

Course Outcomes	<p>At the end of the course, the students should be able to:</p> <ul style="list-style-type: none"> <li>○ Be able to learn database centralized architecture, schema representation, advantages and disadvantages, classification and introduction to DBMS tools.</li> <li>○ Learn high level ER conceptual data modelling, entity, entity sets, attributes and relationships.</li> <li>○ Learn Relational Data Model and Relational Algebra by making use of available SQL application programs.</li> <li>○ Be able to learn Functional dependencies and various normalization forms.</li> <li>○ Learn PL/SQL, Embedded SQL Concepts.</li> </ul>	
UnitNo	Course Content	Hours
<b>UNIT-1</b>	<p><b>Introduction:</b> Database and Database Users, Characteristics of the Database Approach, Actors on the scene, Workers behind the Scene, Advantages of using DBMS, Brief History, Database System Concepts and Architecture: Data Models, Schemas, and Instances, Three Schema Architecture and Data Independence, Database language and interfaces, the database system Environment, Centralized and Client/Server Architectures for DBMS, Classification of Database Management Systems.</p>	<b>12</b>
<b>UNIT - 2</b>	<p><b>Data modelling using the Entity–Relationship(ER) model:</b> High level conceptual data models for database design with an example, Entity types, Entity sets, Attributes and Keys, Relationship types, Relationship sets, Roles and Structural Constraints, Weak Entity Types, ER Diagrams, Naming Conventions.</p> <p><b>Relational Data Model and Relational Algebra:</b> Relation Data Model and Relational Database Constraints, Relation Algebra.</p> <p><b>Queries in SQL:</b>SQL data definition and Data types, specifying constraints in SQL, schema change statements in SQL, Basic queries in SQL, Operators, DDL, DML, DCL, clauses, specifying constraints as Assertions and Triggers, view (Virtual Tables) in SQL, Fragmentation, Indexing</p>	<b>14</b>

<b>UNIT- 3</b>	<p align="center"><b>Functional dependencies and Normalization for Relational Databases</b></p> <p>Informal Design Guidelines for Relational Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definition of 2NF and 3NF, Boyce-Codd Normal Form(BCNF).</p> <p><b>Transaction Processing Concepts:</b> Introduction, Transaction and System Concepts, Desirable properties of transaction.</p>	<b>12</b>
<b>UNIT - 4</b>	<p><b>PL/SQL:</b> Introduction, Datatypes, The PL/SQL syntax, Logical Comparison in PL/SQL, Understanding PL/SQL block structure Identifiers, conditional control, iterative control, cursors- Declaring, opening, closing and fetching from a cursor, stored procedures- syntax, creating, calling and deleting a procedure</p>	<b>12</b>

**Text Books:**

1. Ramez Elmasri & Shamkant B. Navathe, Fundamentals of Database Systems (Sixth Edition), Pearson Education, 2011.
2. Sundarraman, Oracle 9i programming A Primer, 1/e Pearson Education.

**References:**

1. Kahate, Introduction to Database Management System, Pearson Education 2004.
2. Abrahamsi, Silberschatag, Henry. F. Korth, S. Sudarshan, Database System Concepts, Mc. Raw hill.
3. J. D. Ullman, J. Widom. A First Course in Database Systems. Prentice-Hall, 1997.
4. Oracle Press: ORACLE – Computer reference.
5. C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson Education, 2006.
6. Raghu Ram Krishnan, Database Management Systems, Second Edition, McGraw Hill, 2000.

<b>Year</b>	<b>I</b>	<b>Course Code: 24CA208</b>	<b>Credits</b>	<b>4</b>
<b>Semester</b>	<b>II</b>	<b>Course Title: Operating System</b>	<b>Hours</b>	<b>50</b>
<b>Formative Assessment Marks:</b>		<b>Summative Assessment Marks:</b>	<b>Duration of ESA: 03 hrs.</b>	

Course Outcomes	<p>At the end of the course, the students should be able to:</p> <ul style="list-style-type: none"> <li>○ Describe the basics of the operating systems, mechanisms of OS to handle processes, threads, and their communication.</li> <li>○ Analyse the memory management and its allocation policies. ○ Illustrate different conditions for deadlock and their possible solutions. ○ Discuss the storage management policies with respect to different storage management technologies.</li> <li>○ Evaluate the concept of the operating system with respect to UNIX, Linux, Time, and mobile OS.</li> </ul>	
Unit No	Course Content	Hours
UNIT – 1	<p><b>Introduction:</b> Introduction to Operating System, definition, types of operating system, tasks of operating system. Operating System Structures, Components and Services, System Concept, System programs, Virtual machines.</p> <p><b>Process Management:</b> Process concept, Process scheduling, Co operating process, Threads, Inter process communication, CPU scheduling criteria, Scheduling algorithm.</p>	12
UNIT – 2	<p><b>Process synchronization and deadlocks:</b> The critical section problem, Synchronization hardware, Semaphores, Classical problems of synchronization, Critical regions, monitors</p> <p><b>Dead locks</b> –System model, characterization,Dead lock prevention, avoidance and detection, Recovery from dead lock.</p>	12
UNIT – 3	<p><b>Memory Management:</b> Logical and Physical address space, Swapping Contiguous allocation, Paging, Segmentation</p> <p><b>Virtual memory</b> – Demand paging and its performance, page replacement algorithms, Allocation of frames, thrashing, page size and other considerations, Demand Segmentation.</p>	12

<b>UNIT – 4</b>	<p><b>File management (System, Secondary storage structure):</b> File concepts, Access methods, Directory structure, Protection and consistency, semantics, File system structure, Allocation methods, free space management, Directory Implementation, Efficiency and performance, recovery.</p> <p><b>Disk Management (Structure, Disk Scheduling Methods):</b> Disk structure and Scheduling methods, Disk management, Swap – Space management. Protection and Security: Goals of protection, Domain protection, Access matrix security problem, Authentication, One time password, program threats, system threats.</p>	14
-----------------	---	----

**Text books:**

1. Abraham Siberschatz and Peter Bear Galvin, Operating System Concepts, Fifth Edition, Addison – Wesley 19989 (Chapter 1,3.1 , 3.2, 3.3, 3.4, 3.6, 4, 5, 6 (Except 6.8, 6.9),7,8,9,10,11,13,(Except 13.6) 19 (Except 19.6),20(Except 20.8,20.9), 22,23).
2. Nutt: Operating system, 3/e Person Education 2004.

**References:**

1. Milan Milonkovic, Operating System Concepts and Design, II Edition, McGraw Hill 1992.
2. Richard Peterson, Linux – The complete reference, Sixth Edition.
3. William Stallings, “Operating Systems – Internals and Design Principles” Pearson, 6th edition, 2012

<b>Year</b>	I	<b>Course code:</b> 24CA205	<b>Credits</b>	2
<b>Semester</b>	II	<b>Course title:</b> Data structures and algorithms using C++ Lab	<b>Hours</b>	30
<b>Formative assessment marks :</b>		<b>Summative assessment marks :</b>	<b>Duration of ESA : 02 hours</b>	
<b>Course outcomes</b>		At the end of the course the student should be able to: <ul style="list-style-type: none"> <li>· Design the algorithm to analyse the problem</li> <li>· Apply various types of searching and sorting techniques on arrays.</li> </ul>		



	<ul style="list-style-type: none"> <li>· Evaluate the applications of stacks, queues, linked lists and tree</li> </ul>
--	--

### PRACTICE PROGRAMS

1. C++ Program to Calculate Power of a Number
2. C++ Program to Generate Multiplication Table
3. C++ Program to Check Whether a Number is Prime or Not
4. C++ Program to Calculate Sum of Natural Numbers
5. C++ Program to Display Fibonacci Series

### LAB ASSIGNMENTS

1. Write a program to demonstrate linear search
2. Write a program to demonstrate binary search
3. Write a program to demonstrate bubble sort
4. Write a program to demonstrate selection sort
5. Write a program to demonstrate tower of hanoi using recursion
6. Write a program to demonstrate malloc ()
7. Write a program to demonstrate calloc ()
8. Write a program to demonstrate realloc () and free()
9. Write a program to demonstrate stack operations
10. Write a program to demonstrate queue operations
11. Write a program to demonstrate priority queue
12. Write a program to demonstrate singly linked list
13. Write a program to print binary tree and perform in order traversal
14. Write a program to demonstrate prims algorithm
15. Write a program to demonstrate travelling salesman problem

<b>Year</b>	<b>I</b>	<b>Course Code:</b> 24CA207	<b>Credits</b>	<b>2</b>
<b>Semester</b>	<b>II</b>	<b>Course Title:</b> Database Management Lab	<b>Hours</b>	<b>3 Hours</b>
<b>Course Pre-requisite, if any</b>		NA		

<b>Formative Assessment Marks: 20</b>	<b>Summative Assessment Marks: 30</b>	<b>Duration of ESA: 03 hrs.</b>
<b>Course Outcomes</b>	At the end of the course, the students should be able to: <ul style="list-style-type: none"> <li>o Creation and Manipulation of Database</li> <li>o Execute a single line query and group functions.</li> <li>o Execute DDL DML, DCL and TCL commands</li> <li>o Implement the Nested Queries and Join operations in SQL</li> <li>o Create views for a particular table</li> </ul>	

## Part A

### TERM WORK – I

#### COLLEGE DATABASE

##### Consider the College database:

Course (courseid:number, coursename:string, duration:number)

Student (id:number, fname:string, mname:string, lname:string, address:string, mobile:number, courseno:number)

Subject(sid:number, sname:string, cno:number)

Faculty(fid:number, fname:string, lname:string, sno:number)

1. Draw ER and Schema Diagram
2. Create above tables by properly specifying primary and foreign keys.
3. Insert at least 5 values to all tables.
4. Find Courses starts from BSC
5. Display Student name and his/her course
6. Which subject is handled by Faculty SheetalSoni?
7. Find Faculties name, subjects teaches for BCA.
8. Count number of students course wise.

### TERM WORK – II

#### CUSTOMER – ORDER DATABASE

##### Consider the Customer Order Product Database:

Customer (custid: number, fname:String, lname:String, city:String, state: string, mobile:number) Product (productid:varchar, productcat:varchar, product:varchar, price:decimal) Ordertab (orderid:number, orderdate:date,custid:number, pid:varchar, qty:number) 1) Draw ER and Schema Diagram

- 2) Create above tables by properly specifying primary and foreign keys.
- 3) Insert at least 5 values to all tables.
- 4) Find customer name start with so
- 5) Find product wise product count.

- 6) Display the count of product categories.
- 7) Display fname, lname of the customer with product and amount.
- 8) Update the address of customer Umesh from belagavi to Pune.

### **TERM WORK – III**

#### **Movie Database**

##### **Consider the Movie Database:**

Actor (Act\_id, Act\_Name, Act\_Gender)

Director (Dir\_id, Dir\_Name, Dir\_Phone)

Movies (Act\_id, Mov\_id, Mov\_Title, Mov\_Year, Mov\_Lang, Dir\_id)

Movie\_cast (Act\_id, Mov\_id, Role)

Rating (Mov\_id, Rev\_Stars)

1. Draw ER and schema diagram for database.
2. Create the above tables by properly specifying the primary keys and foreign keys.
3. Enter at least five tuples for each relation.
4. List the titles of all movies directed by 'Hitchcock'.
5. Find the maximum rating of movie
6. Display names of actors acted in more than one movie.
7. List all actors who acted in a movie between 2000 and 2015
8. Update rating of all movies directed by 'Steven Spielberg' to 5

### **TERM WORK – IV**

#### **COMPANY DATABASE**

##### **Consider the Company Database:**

Employee(Fname:string, Minit:string, Lname:string, id:number, Bdate:date, Address:string, Sex:character, Salary:decimal, Dno:int)

Department(Dname:string, Dnumber:int, id:number,

mgr\_start\_date:date) Project(Pname:string, Pnumber:int,

Plocation:string, Dnum:int)

works\_on(id:number, Pno:int, Hours:decimal)

1. Draw ER and Schema Diagram
2. Create above tables by properly specifying primary and foreign keys.
3. Insert at least 5 values to all tables.
4. Select employees working for department 1 and 3
5. Retrieve bdate ,address of employee whose name is 'Amar B Shinde'
6. Retrieve name, address of employees who are working for Research department
7. for every

project located in Bangalore list project name, controlling department dnumber, department manager last name ,address, bdate.

8. Display name, project name and number of hours employee working on project

## **Part B**

### **PLSQL Programs**

1. Write a PL/SQL program to find sum of two numbers.
2. Write a PL/SQL program for finding Multiples of 5
3. Write a PL/SQL program to check weather given number is Prime or not.
4. Write a PL/SQL program to finding factorial of given number
5. Write a PL/SQL program to generate reverse for given number.
6. Write a PL/SQL program to generate Fibonacci series.
7. Write a PL/SQL program to print a string in a letter by letter format
8. Write a PL/SQL program to find given number is even or odd
9. Write a PL/SQL program for display the Multiplication Tables up to given number
10. Write a PL/SQL program for inserting a row into vender table