RANI CHANNAMMA UNIVERSITY
BELAGAVI

REVISED CURRICULUM FRAMEWORK FOR
UNDER GRADUATE COURSE

STRUCTURE & SYLLABUS OF BACHELOR OF COMPUTER APPLICATION

BCA

1ST TO 2ND Semesters

w.e.f.

Academic Year 2024-25 and Onwards

Submitted by

Chairman,
Board of Studies (UG),
Bachelor of Computer Science,

PROGRAM OUTCOMES:

By the end of the program the following outcomes will be achieved by the students:

- Discipline knowledge: Acquiring knowledge on basics of Computer Science and ability to apply to design principles in the development of solutions for problems of varying complexity

- Problem Solving: Improved reasoning with strong mathematical ability to Identify, formulate and analyse problems related to computer science and exhibiting a sound knowledge on data structures and algorithms.

- Design and Development of Solutions: Ability to design and development of algorithmic solutions to real world problems and acquiring a minimum knowledge on statistics and optimization problems. Establishing excellent skills in applying various design strategies for solving complex problems.

- Programming a Computer: Exhibiting strong skills required to program a computer for various issues and problems of day-to-day applications with thorough knowledge on programming languages of various levels.

- Application Systems Knowledge: Possessing a sound knowledge on computer application software and ability to design and develop app for applicative problems.

- Modern Tool Usage: Identify, select and use a modern scientific and IT tool or technique for modelling, prediction, data analysis and solving problems in the area of Computer Science and making them mobile based application software.

- Communication: Must have a reasonably good communication knowledge both in oral and writing.

- Project Management: Practicing of existing projects and becoming independent to launch own project by identifying a gap in solutions.

- Ethics on Profession, Environment and Society: Exhibiting professional ethics to maintain the integrality in a working environment and also have concern on societal impacts due to computer-based solutions for problems.

Revised Course Structure and Syllabus

		I Semester BCA w.e.f 2024-25 and onwards							
Part	Course Code	Subject Name	Teaching Hrs / week	Practical Hrs / week	Examination				Credits
					Exam Duration Hrs	Marks			
						IA	Theory / Practical	Total	
Part I AECC	SEPBCAAEC 1.1	Language 1	4	-	3	20	80	100	3
	SEPBCAAEC 1.2	Language 2	4	-	3	20	80	100	3
Part II DSC	SEPBCAT1.3	Computer Concepts and C Programming	4	-	3	20	80	100	3
	SEPBCAT1.4	Database Management System	4	-	3	20	80	100	3
	SEPBCAT1.5(a) / SEPBCAT1.5(b)*	Financial Accounting / Mathematics-I	4	-	3	20	80	100	3
	SEPBCAP1.6	C Programming Lab	-	4	3	10	40	50	2
	SEPBCAP1.7	Database Management System Lab	-	4	3	10	40	50	2
	SEPBCAP1.8	Office Automation lab	-	4	3	10	40	50	2
Part III SEC	SEPBCAAEC 1.9	Compulsory 1 (Constitutional Values)	2	-	2	10	40	50	2
Total			22	12				700	23

Note- * BCADSC 1.5(a) – for students from Science Background / BCADSC 1.5(b)* for students from Non- Science Background

		II Semester BCA w.e.f 2024-25 and onwards							
Part	Course Code	Subject Name	Teaching Hrs / week	Practical Hrs / week	Examination				Credits
					Exam Duration Hrs	Marks			
						IA	Theory / Practical	Total	
Part I AECC	SEPBCAAEC 2.1	Language 1	4	-	3	20	80	100	3
	SEPBCAAEC 2.2	Language 2	4	-	3	20	80	100	3
Part II DSC	SEPBCAT 2.3	Data Structures and Algorithms	4	-	3	20	80	100	3
	SEPBCAT 2.4	Numerical and Statistical Methods	4	-	3	20	80	100	3
	SEPBCAT 2.5	Object Oriented Programming Using Java	4	-	3	20	80	100	3
	SEPBCAP 2.6	Data Structures and Algorithms Lab	-	4	3	10	40	50	2
	SEPBCAP 2.7	Numerical and Statistical Methods Lab	-	4	3	10	40	50	2
	SEPBCAP 2.8	Object Oriented Programming Using Java Lab	-	4	3	10	40	50	2
Part III SEC	SEPBCAAEC 2.9	Compulsory 2 (Constitutional Values)	2	-	2	10	40	50	2
Total			22	12				700	23

Year	I	Course Code: SEPBCAT1.3	Credits	3
Semester	I	Course Title: Computer Concepts and C Programming	Hours	48
Course Pre-requisite, if any	NA			
Formative Assessment Marks: 20	Summative Assessment Marks: 80		Duration of ESA: 03 hrs.	
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	Introduction: Computer, data processing, characteristic features of computers, Basic operations performed by computers, basic organization of computer system Number systems: Different types- decimal, binary, octal, and hexadecimal number systems, Conversion from decimal to binary and vice- versa Software: Software and its relationship with hardware, types of software, system software, application software, firmware, middleware, Computer languages: Machine Level, Assembly Level & High Level Languages, Translator Programs – Assembler, Interpreter and Compiler	10
Unit-2	Steps for Problem Solving: Flowcharts, Algorithms; Examples of flow charts and algorithms- Largest of three numbers, reversing the digits of an integer, GCD of two integers, generating prime numbers, computing n Fibonacci numbers. Introduction to C Programming: Features of C; Structure of a C Program with Examples; Creating and Executing a C Program; Compilation process in C. C Programming Basic Concepts: C Character Set; C tokens - keywords, identifiers, constants, and variables; Data types; Declaration & initialization of variables; Symbolic constants.	10
Unit-3	Input and output with C: Formatted I/O functions - printf and scanf, control strings and escape sequences, output specifications with printf functions; Unformatted I/O functions to read and display single character and a string - getchar, putchar, gets and puts functions. Character handling library functions - toascii, toupper, tolower, isalpha, isnumeric C Operators & Expressions: Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators, Operator Precedence and Associatively; Evaluation of arithmetic expressions; Mathematical library functions in C using math.h; Type casting & conversion.	8
Unit-4	Control Structures: Decision making Statements - Simple if, if_else, nested if_else, else_if ladder, Switch Case, goto, break & continue statements; Looping Statements - Entry controlled and exit controlled statements, while, do-while, for loops, Nested loops.	10

	Strings: String constant and variables, Declaration and initialization of string, Input/output of string data. String Library Functions: strlen, strcat, strcmp, strcpy, strrev.	
Unit-5	Arrays: Definition, types, initialization, processing an array, passing arrays to functions, Array of Strings. User defined functions: Definition, types of user defined functions, prototype, Local and global variables, passing parameters, recursion Structures & Union: Definition of Structure, declaring Structure, accessing Structure elements, array of Structure, Nesting of structure. Definition of Union, declaring and using Union, Difference between Structure & Union.	10

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

Year	I	Course Code: SEPBCAT1.4	Credits	3
Semester	I	Course Title: Database Management System	Hours	48
Course Pre-requisite, if any	NA			
Formative Assessment Marks: 20	Summative Assessment Marks: 80		Duration of ESA: 03 hrs.	
Course Outcomes	At the end of the course, the students should be able to: o Explain the various database concepts and the need for database systems. o Identify and define database objects, and enforce integrity constraints on a database using DBMS. o Demonstrate a Data model and Schemas in RDBMS. o Identify entities and relationships and draw an ER-diagram for a given real-world problem. o Convert an ER diagram to a database schema and deduce it to the desired normal form. o Formulate queries in Relational Algebra, Structured Query Language (SQL) for database manipulation.			

Unit No	Course Content	Hours
Unit-1	Introduction to Database System Concepts and Architecture Databases and Database Users, Characteristics of the Database Approach, Actors on the Scene, Advantages of Using a DBMS Data Models, Schemas and Instances, DBMS Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment.	8
Unit-2	Data Modelling Using the Entity-Relationship Model Entity Types, Entity Sets, Attributes, and Keys, Relationship Types, Relationship Sets, Roles, and Structural Constraints, Weak Entity Types, ER Diagrams, Naming Conventions and Design Aspects.	10
Unit-3	Relational Data Model, Relational Constraints, and Relational Algebra Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Basic Relational Algebra Operations. Examples of Queries in Relational Algebra.	10
Unit-4	Normalization- Functional Dependencies, Transitive and Multivalued Dependencies, First Normal Form, Second Normal Form, Third Normal Form, BCNF	10
Unit-5	Introduction to SQL: SQL and its features, SQL Data types, DDL: Create tables (with constraints), Alter tables, DML: Insert, Update, Delete Operations, Queries: Simple Queries, select with WHERE clause, ORDER BY clause and aggregate functions, Using operators like BETWEEN, IN, ANY, ALL and computations in queries, Simple Queries using all clauses. More on SQL: Joins, types of joins, queries with GROUP BY clause, queries with HAVING clause, views and their relevance, creating and using views, simple example queries for joining tables and using GROUP BY and HAVING clause.	10

TextBooks:

1. Fundamentals of Database Systems, Ramez Elamassri, Shankant B. Navathe, 7th Edition, Pearson, 2015.
2. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill, 2010.

References:

1. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010.
2. Introduction to Database System, C J Date, Pearson, 1999.
3. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition,
4. McGraw Hill, 2002

Year	I	Course Code: SEPBCAT1.5(a)	Credits	3
Semester	I	Course Title: Financial Accounting	Hours	48
Course Pre-requisite, if any		Students should have studied Science in PU/12th level		
Formative Assessment Marks: 20		Summative Assessment Marks: 80	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course, the students should be able to: o Understand accounting, branches of accounting and its role. o Know the concepts of accounting and accounting process. o Maintenance of different accounting books. o Preparation of final accounts.		

Unit No	Course Content	Hours
Unit-1	Introduction to Accounting: Meaning of book-keeping, accounting and accountancy. Accounting – objectives, scope & functions. Book Keeping Vs Accounting. Users of accounting information. Systems of Accounting – Single entry system & Double entry system, differences. Branches of Accounting. Accounting equation. Important accounting terms.	10
Unit-2	Accounting Principles – Concepts & Conventions: Generally Accepted Accounting Principles, Accounting Policies, Principles vs Polices. Accounting cycle. Classification of Accounts – British System & American System. Rules of Accounting, i.e. Debit & Credit rules under British System & American System. Journal – meaning, features, format & journalising. Ledger – meaning, features, format, ledger posting, balancing a ledger account. Journal vs Ledger. Preparation of personal ledger account.	09
Unit-3	Subsidiary Books: Meaning, features & advantages. Types – Purchase Book, Purchase Returns Book, Sales Book, Sales Returns Book, Cash Book (Simple Cash Book & Three Column Cash Book) Meaning of terms – discount, cash discount, trade discount, invoice, debit note, credit note, contra entry, voucher, receipt, pass book, etc.	09
Unit-4	Bank Reconciliation Statement: Meaning & need. Reasons for difference between cash book balance and pass book balance. Preparation of BRS. Advantages.	10
Unit-5	Final Accounts: Meaning, need & objectives. Trial Balance – meaning, objectives, format & preparation. Meaning & objectives of preparation of Trading Account, Profit & Loss Account and Balance Sheet. Meaning of Gross Profit, Gross Loss, Net Profit, Net Loss, Depreciation, Bad Debts, Outstanding Expenses, Prepaid Expenses, Capital, and Assets & Liabilities. Simples problems on preparation of Trading Account, Profit & Loss Account and Balance Sheet [Sole Trading Concerns only]	10

Text books:

1. Book Keeping & Accountancy by M. B. Kadkol
2. Accountancy by B. S. Raman
3. Book Keeping & Accounting by V. A. Patil & J. S. Korlahalli

References:

1. Dr. R.K. Mittal & M.R. Bansal, Financial Accounting, VK Publications.
2. Anil Chowdhry, Fundamentals of Accounting & Financial Analysis, Pearson Education.
3. Maheshwari&Maheshwari, An Introduction to Accountancy, 11th Edition, Vikas Publishing House.
4. Jane Reimers, Financial accounting, Pearson Education
5. RajniPreetiHiroSofat, Basic Accounting, PHI Additional Reading:
6. Accounting for management, Bhattacharya & Deaden, Paperback Edition, Vikas 1986
7. Financial Accounting (Part I and Part II), R.L Gupta & V.K Gupta
8. Maheshwari S.N., Principles of Management Accounting, Sultan Chand & Sons,
9. Accounting Principal, Antony & Reece, Sixth Edition

Year	I	Course Code: SEPBCAT1.5(b)	Credits	3
Semester	I	Course Title: Mathematics –I	Hours	48
Course Pre-requisite, if any		Students should have studied Commerce in PU/12th level		
Formative Assessment Marks: 20		Summative Assessment Marks: 80	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course, the students should be able to: o To perform basic arithmetic operations and roots on complex numbers. o Apprehend types of Sequences & Series and relationship between them. o Understand Expansion of Binomial Expression and coefficient, Pascal's triangle and find the any term in the expansion. o Explore the basic concepts of trigonometry, including sine, cosine, and tangent functions and Standard Angles, Allied Angles. o Understand the basic concepts of Distance Formulae, Section Formulae, Slope of a Line and Angle between Two Lines. Various Forms of Equations of a Line		

Unit No	Course Content	Hours
Unit-1	Complex Numbers: Complex Numbers; Conjugate of a Complex Number; Modules of a Complex Number; Geometrical Representation of Complex Number; De Moivere'sTheorem; nth Roots of a Complex Number.	10
Unit-2	Sequence And Series: Arithmetic Progression (A P),Arithmetic Men (A M), Geometric Progression (G P),General Term of a G P, Sum of N Terms of a G P. Arithmetic and Geometric Series, Infinite G P and its Sum, Geometric Mean(G M) Relation Between A M and G M	10
Unit-3	Binomial Theorem: Statement of the Binomial Theorem For Positive Integral Indices, General And Middle Term in Binomial Expansion, Simple Applications. Quadratic Equations: Solution Of Quadratic Equations by Factor Method, Complete Square Method, And Discriminant Method, Relation Of The Roots.	10
Unit-4	Introduction To Trigonometry: Trigonometry Ratio's, Trigonometric, Functions of Standard Angles,Allied Angles, CompoundAngles, Multiple & Sub Multiple Angles, Transformation Formula	9
Unit-5	Co-Ordinate Geometry: Distance Formulae, Section Formulae, Shifting Of Origin. Slope of a Line And Angle Between Two Lines. Various Forms of Equations of A Line: Parallel To Axes, Point Slope Form, Slope-Intercept Form, Two-Point Form, Intercepts Forms And Normal Form, General Equation of a line. Equation of Family of Lines Passing Through The Point of Intersection of Two Lines. Distance of A Point From A Line.	9

Text Books

1. 11th& 12th NCERT Mathematics books.
2. B. S. Grewal, Elementary engineering mathematics, Khanna publishers

References:

1. S. L. Loney, M. A., the elements of coordinate geometry, scholarly publishing office, university of Michigan library
2. S. L. Loney, M. A., plane trigonometry, scholarly publishing office, university of Michigan libraray

<table>
<tr><td>Year</td><td>I</td><td>Course Code: SEPBCAP1.6</td><td>Credits</td><td>2</td></tr>
<tr><td>Semester</td><td>I</td><td>Course Title: C Programming Lab</td><td>Hours</td><td>48</td></tr>
<tr><td colspan="2">Course Pre-requisite, if any</td><td colspan="3">NA</td></tr>
<tr><td colspan="2">Formative Assessment Marks: 10</td><td>Summative Assessment Marks: 40</td><td colspan="2">Duration of ESA: 03 hrs.</td></tr>
<tr><td colspan="2">Course Outcomes</td><td colspan="3">At the end of the course, the students should be able to:
o Get familiar with working in Linux environment using basic shell commands such as ls, cd, mv, mkdir, rm, cat,etc.
o Write algorithms and design flowcharts to solve problems
o Use vi/gedit/any terminal editor for writing C programs
o Use Linux-Open Source OS for debugging and executing C programs using gcc/similar compiler available with Linux.</td></tr>
</table>

Lab Assignments (To be implemented using C Programming Language)

Part A

1. Write a program to read radius of a circle and to find and display area and circumference
2. Write a program to read Principal amount, Time and Rate and calculate Simple Interest.
3. Program to read three numbers and display the largest of the three.
4. Write a program to function as a basic calculator; it should ask the user to input what type of arithmetic operation he would like, and then ask for the numbers on which the operation should be performed. The calculator should then give the output of the operation. Use switch. Error message should be reported, if any attempt is made to divide by zero.
5. Write a program to find the roots of quadratic equation (Demonstration of else-if ladder)
6. Write a program to read marks scored by a student in n subjects and find and display the average of all marks
7. Write a program to demonstrate basic mathematical library functions defined in math.h
8. Write a program to find HCF (GCD) of two numbers.
9. Write a program that accepts a number ‘n’, and prints all prime numbers between 1 to n.
10. Write a program to read a number, find the sum of its digits, reverse the number and check if it is a palindrome. Display appropriate message.
11. Write a program to read numbers from keyboard continuously till the user presses 999 and to find the sum of only positive numbers. Display appropriate error messages.
12. Write a program to read percentage of marks and to display appropriate message specifying class obtained. (demonstration of Switch Case statement)
13. Write a program to read and concatenate two strings without using library function
14. Write a program to print sum of even numbers and sum of odd numbers from array of integers which are to be inputed.
15. Write a program to read a list of numbers, print it, then remove duplicate elements from the list and print the modified list. Use single dimensional Array

Part B

1. Write a program to swap(interchange) two numbers without using a temporary variable
2. Write a program to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters using character functions
3. Write a program to read a string, reverse it, concatenate the original string with its reverse, and print length of the original string and the concatenated string using built in functions
4. Write a program to read a string and find the length of a string without using built in function
5. Write a program to generate and display first n values in the Fibonacci sequence using recursive function
6. Write a recursive function calculate factorial of a given integer, n.

7. Write a program to read elements in a square matrix, display it in the form of a matrix and find and print its trace
8. Write a program to read two matrices and perform addition and subtraction on them
9. Write a program to read, display and multiply two m x n matrices using functions.
10. Write a program to check and display if a number is prime by defining isprime( ) function
11. Write a program using functions that takes in three arguments - a start temperature (in Celsius), an end temperature (in Celsius) and a step size. Print out a table that goes from the start temperature to the end temperature, in steps of the step size; converting each Celsius to Fahrenheit.
12. Write a C program to create array of structure which stores Roll No, Name and Average marks of students. Accept n students data and print it in proper format.
13. Write a C program to illustrate difference between structure and union by defining emp_no , emp_name, salary as members and display the size of the defined structure

<table>
<tr><td>Year</td><td>I</td><td>Course Code: SEPBCAP1.7</td><td>Credits</td><td>2</td></tr>
<tr><td>Semester</td><td>I</td><td>Course Title: Database management System Lab</td><td>Hours</td><td>48</td></tr>
<tr><td colspan="2">Course Pre-requisite, if any</td><td colspan="3">NA</td></tr>
<tr><td colspan="2">Formative Assessment Marks: 10</td><td>Summative Assessment Marks: 40</td><td colspan="2">Duration of ESA: 03 hrs.</td></tr>
<tr><td colspan="2">Course Outcomes</td><td colspan="3">At the end of the course, the students should be able to:
o Creation and Manipulation of Database
o Execute a single line query and group functions.
o Execute DDL DML, DCL and TCL commands
o Implement the Nested Queries and Join operations in SQL
o Create views for a particular table</td></tr>
</table>

Lab Assignments(To be implemented in SQL)

Case Study 1:
Basic EMPLOYEE DATABASE

The EMP detail databases have a table with the following attributes. The primary keys are underlined.
EMPLOYEE(EmpNo: int, name: string, dob: date, PhNo: int)

a. Create the above table.
b. Remove the existing attributes from the table.
c. Change the date type of EmpNo from integer to string.
d. Add a new attribute Date of Joining(DOJ) to the existing table.
e. Enter five tuples into the table.
f. Display all the tuples of the employee table.

Case Study 2:
Student Database
Consider the following database of student's enrolment in courses and books adopted for each course.
STUDENT (regno: string, name: string, major: strong, bdate: date)
COURSE (course-no: int cname: string, dept: string)
ENROLL (reg-no: string, course-no: int, sem: int, marks: int)
BOOK-ADOPTION (course-no: int, sem: int, book-isbn: int)
TEXT (book_isbn: int, book-title: string, publisher: string, author: string)

a. Create the above tables by properly specifying the primary keys and the foreign keys
b. Enter at least five tuples for each relation.
c. Demonstrate how you add a new textbook to the database and make this book be adopted by some department.
d. Produce a list of textbooks (include Course-no, book_isbn, book-title) in alphabetical order for courses offered by the 'Compute Science' department that use more than two books.
e. List any department that has all its adopted books published by a specific publisher.

Case Study 3:
Consider the following database for BANK.
BRANCH (branch-name: string, branch-city: string, assets: real)
ACCOUNT (accno: int, branch-name: string, balance: real)
DEPOSITOR (customer-name: string, accno: int)

CUSTOMER (customer-name: string, customer-street: string, customer-city: string)
LOAN (loan-no: int, branch-name: string, amount: real)
BORROWER (customer-name: string, loan-no: int)

a. Create the above tables by properly specifying the primary keys and foreign keys.
b. Enter at least five tuples for each relation
c. Find all the customers who have at least two accounts at the main branch.
d. Find all customer who have an account at all the branches located in a specific city
e. Demonstrate how to delete all account tuples at every branch located in specific city.

Case Study 4:

Railway Reservation System
TRAIN(Train_no,Train_name,Start_Place, Destination)
AVAILABILITY(Train_no,Class,Start_Place,Destination,No_of_seats)

1. Create a view sleeper to display train no, start place, destination which have sleeper class and perform the following
   - insert new record
   - update destination="Delhi" where train no="DLD19"
   - delete a record which has train no="KKE55"
2. Create view details to display train no, train name, class
3. Create view total_seats to display train number, start place, use count function to no of seats , group by start place and perform the following
   - insert new record
   - update start place="Hubli" where train no="JNS10"
   - delete last row of the view
4. Rename view sleeper to class
5. Delete view details.

Case Study 5:

Consider the following relations for an order processing database application in a company.
CUSTOMER (CUST #: INT, CNAME: STRING, CITY: STRING)
ORDER (ORDER #: INT, ODATE: DATE, CUST #: INT, ORD-AMT: INT)
ITEM (ITEM #: INT, UNIT PRICE: INT)
ORDER - ITEM (ORDER #: INT, ITEM #: INT, QTY: INT)
SHIPMENT (ORDER #: INT, WAREHOUSE#: INT, SHIP-DATE: DATE)
WAREHOUSE (WAREHOUSE #: INT, CITY: STRING)
(i) Create the above tables by properly specifying the primary keys and the foreign keys.
(ii) Enter at least five tuples for each relation.
(iii) Produce a listing: CUSTNAME, #of orders, AVG_ORDER_AMT, where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer.
(iv) List the order# for orders that were shipped from all the warehouses that the company has in a specific city.
(v) Demonstrate how you delete item# 10 from the ITEM table and make that field null in the ORDER_ITEM table.
(vi) Generate suitable reports.
(vii) Create suitable front end for querying and displaying the results.

Year	I	Course Code: SEPBCAP1.8	Credits	2
Semester	I	Course Title: Office Automation Lab	Hours	48
Course Pre-requisite, if any		NA		
Formative Assessment Marks: 10		Summative Assessment Marks: 40	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course, the students should be able to: o Gain Knowledge about Word processor, Spreadsheets and Presentation		

Assignments (To be Implemented using any Open Source Software)

Part A

1. Word Processor assignment to demonstrate usage of Page Setup, Page Background and Paragraph option of Page Layout tab by writing the description about Computer and its characteristics.

2. Word Processor assignment to demonstrate Bullets and Numbering, Headers and footers.

3. Word Processor assignment to demonstrate usage of tables and encryption by preparing the time table.

4. Word Processor assignment to design a pamphlet for the advertisement of your college features by making use of Picture ,ClipArt , Shapes and WordArt options.

5. Word Processor assignment to demonstrate usage of mail merge by creating a letter to invite your parents and friends for the annual day event.

6. Create a power-point presentation with minimum 5 slides and demonstrate the following.
   a) Layout option b) Insertion of date, time and slide numbers c) Insertion of Symbols

7. Create a power-point presentation with minimum 5 slides and demonstrate the following. a) Themes b) Transitions c) Animation

8. Create a power-point presentation with minimum 5 slides and demonstrate the following. a) Rehearse Timings b) Narrations c) Slide Sorter

Part B

1. Spread sheet assignment to create a pay slip to generate salary of employee with the following details EMP No, BASIC, DA, HRA, GROSS, PF,NET SALARY.

2. Spread sheet Assignment to create SSLC Marks card and demonstrate charts. Use the following details SNO, NAMES, KAN, HIND, ENG, MAT, SCI, SOC, TOTAL, AVG, MIN, MAX, RESULT, and CLASS.

3. Spread sheet assignment to generate bill and to demonstrate various statistical

functions with the following details GRAIN ITEM, UNITS (KG), PRICE, AMOUNT, GRADE.

4. Demonstrate the following in Spread sheet as directed
   a. Create a notepad file as per the following fields:
      Sl.no. name Sub1 Sub2 Sub3 Sub4 Sub5 total % grade
   b. Import this notepad file into Spread sheet using „data from text option.
   c. Grade is calculated as,
      i. If %>=90, then grade A
      ii. If %>=80 and <90, then grade B
      iii. If %>=70 and <80, then grade C
      iv. If %>=60 and <70, then grade D
      v. If %<60, then grade F

5. Demonstrate Sorting and Filtering of Data by using the following details.
   Slno ItemName Rate Quantity Amount Discount (10%) Final Amount

6. Demonstrate Data validation , LookUp and VLookUp Functions.

7. Demonstrate Pivot tables and Pivot charts.

SEMESTER II

Year	I	Course Code: SEPBCAT2.3	Credits	3
Semester	II	Course Title: Data Structures and Algorithms	Hours	48
Course Pre-requisite, if any	Knowledge of C Programming			
Formative Assessment Marks: 20	Summative Assessment Marks: 80		Duration of ESA: 03 hrs.	
Course Outcomes	At the end of the course the student should be able to: o Design the algorithm to analyse the problem o Differentiate between iteration and recursion and apply recursive definition for problem solving o Apply various types of searching and sorting techniques on arrays o Evaluate the applications of stacks, queues, linked lists and tree			

Unit No	Course Content	Hours
Unit-1	**Pointers in C**: Understanding pointers - Declaring and initializing pointers, accessing address and value of variables using pointers; Pointers and Arrays; Pointer Arithmetic; Advantages and disadvantages of using pointers **Dynamic memory allocation**: Static & Dynamic memory allocation; Memory allocation and de-allocation functions - malloc, calloc, realloc and free.	10
Unit-2	**Recursion**: Definition, Types of recursions, Recursion Technique Examples - GCD, Binomial coefficient nCr, Towers of Hanoi. Comparison between iterative and recursive functions. **File Management in C**: Defining, declaring a file, Opening & Closing File, Input & Output Operations on Files, File error handling functions. **Introduction to data structures**: Definition; Types of data structures - Primitive & Non-primitive, Linear and Non-linear; Operations on data structures.	10
Unit-3	**Sorting and Searching:** Representation of linear arrays in memory, Traversing linear arrays; Inserting and deleting elements; **Sorting**: Selection sort, Bubble sort, Quick sort, Merge sort, Insertion sort **Searching:** Sequential Search, Binary search; Iterative and Recursive searching.	10
Unit-4	**Stacks**: Definition and Representation of stacks; Operations on stacks – Push, Pop; Applications of stacks; Infix, postfix and prefix notations; Conversion from infix to postfix using stack; Evaluation of postfix expression using stack; Application of stack in function calls. **Queues**: Definition and Representation of queues; Types of queues - Simple queues, Circular queues, Double ended queues, Priority queues; Operations on Simple queues.	9
Unit-5	**Linked list**: Definition and Representation of linked list, Types of linked lists - Singly linked list, Doubly linked list, Circular linked list Doubly Circular Linked list; Representation of Linked list in Memory; Operations on Singly linked lists – Traversing, Searching, Insertion, Deletion;	9

	Trees: Definition; Tree terminologies –node, root node, parent node, ancestors of a node, siblings, terminal & non-terminal nodes, degree of a node, level, edge, path, depth; Binary tree: Type of binary trees - strict binary tree, complete binary tree, binary search tree. Traversal of binary tree; preorder, in order and post order traversal.	

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

<table>
<tr><td>Year</td><td>I</td><td>Course Code: SEPBCAT2.4</td><td>Credits</td><td>3</td></tr>
<tr><td>Semester</td><td>II</td><td>Course Title:
Numerical and Statistical Methods</td><td>Hours</td><td>48</td></tr>
<tr><td colspan="2">Course Pre-requisite, if any</td><td colspan="3"></td></tr>
<tr><td colspan="2">Formative Assessment Marks: 20</td><td>Summative Assessment Marks: 80</td><td colspan="2">Duration of ESA: 03 hrs.</td></tr>
<tr><td colspan="2">Course Outcomes</td><td colspan="3">At the end of the course the student should be able to:
o Learn how to obtain numerical solution of nonlinear equations using Bisection, Newton – Raphson and fixed-point iteration methods.
o Solve system of linear equations numerically using direct and iterative methods
o Develop a framework for estimating and predicting the different sample of data for handling the uncertainties.
o Understand the various approaches dealing the data using theory of probability</td></tr>
</table>

Unit No	Course Content	Hours
Unit-1	Introduction and solutions of algebraic and transcendent equations, methods of finding solution of non- linear equations- Bisection method, False Position method, Secant method, Fixed point iteration method and Newton-Raphson method. General discussion on convergence of these methods (No Mathematical derivations)	10
Unit-2	Introduction and methods of solving of solving system of linear equations - Gauss elimination method, Gauss- Jordan, LU Decomposition method and Gauss-Seidal iteration method, Eigen values and Eigen vectors of a square matrix.	10
Unit-3	Origin and development, Definition, Importance and scope of business Statistics, Meaning and definition of data, Methods of data collection. Types of data proportions, ratios and rates; building, cleaning and administering databases in SPSS. Significance of diagrams and graphs, Types of diagrams – one dimensional or Bar Diagrams, Two dimensional or area diagrams, pictograms and cartograms. Graphs of frequency distribution-Histogram, frequency polygon, Frequency curve, gives or cumulative frequency curves.	10
Unit-4	Definition of averages, objectives of averages, requisites of ideal averages. Types of averages- A mean, median, Mode, Harmonic mean, Geometric Mean – Definition computation, merits and demerits,Application in Business. Definition and properties of Ideal Measure of dispersion, Absolute and Relative Measures of dispersion- Range and co-efficient of range, Quartile and co-efficient of Q.D., Average Deviation(AD) and co-efficient of A.D., Standard Deviation and co-efficient of S.D. and co-efficient of variation.	9
Unit-5	Properties of Probability, Equally likely event, mutually exclusive, Independent and Conditional Probability. Types of Probability Distributions - Bernoulli, Binomial, Poisson, fitting a Binomial and Poisson distribution, Normal distribution and problems.	9

Text Books:

1. S.S.Sastri, Introductory Methods of Numerical Analysis, PHI(NewDelhi)2001.
2. Balaguruswamy E, (1988), Computer Oriented Statistical and Numerical Method, Macmillan India Ltd.
3. Medhi J.1992,Statistical Methods (An Introductory Text),New Age International.
4. Business Statistics by-J K Sharma, Pearson Publication.

References:

1. M.K.Jain, S.R.K.Iyenger and R.K.Jain, Numerical Method for Scientific and Engineering Computation, Wiley Eastern (1998).
2. V.RajaRaman Computer oriented numerical methods, PHI Publication
3. Gupta S.C. and Kapoor V.K. 2005 Fundamentals of Mathematical Statistics, S.Chand and Sons, New Delhi.
4. Gupta S.C. and Kapoor V.K.2005 Fundamentals of Applied Statistics, S.Chand and Sons, New Delhi.

Year	I	Course Code: SEPBCAT2.5	Credits	3
Semester	II	Course Title: OOPS Using Java	Hours	48
Course Pre-requisite, if any				
Formative Assessment Marks: 20		Summative Assessment Marks: 80	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course the student should be able to: o Understand the features of Java and different tools available in Java Environment. o Write, compile, and execute Java programs that may include basic data types and control flow constructs and how type casting is done. o Identify classes, objects, members of a class and relationships among them needed for a specific problem and demonstrate the concepts of polymorphism and inheritance o Demonstrate programs based on arrays and strings. o Demonstrate programs based on interfaces and threads. o Explain the benefits of JAVA’s Exceptional handling mechanism compared to other Programming Languages.		

Unit No	Course Content	Hours
Unit-1	Concept - Object, classes, abstraction, encapsulation, inheritance, polymorphism .History of Java, features of Java, Java Environment - JDK, API, Java Runtime environment, JVM, Java program structure, Identifiers and Keywords, Data types and typecasting, Variables, operators, Java coding conventions, Expressions, Control structures, Decision making statements, Arrays and its methods, command line arguments.	10
Unit-2	Java classes, Define class with instance variables and methods, Object creation, Accessing member of class, argument passing, Constructors, Method overloading, constructor overloading, Static data, Static methods, Static blocks, This keyword, Garbage collection & finalize() method, Arrays.	10
Unit-3	String -String Arrays, String Methods, String Buffer, Inheritance: Super class & subclass, abstract method and classes, method overriding, final keyword, super keyword, dynamic method dispatch.	9
Unit-4	Packages and Interfaces: Implementing interfaces, user defined interfaces; modifiers & access control (Default, public, private and protected), user defined packages, Importing classes Exception handling: Types of Exceptions, try, catch, finally, throw, throws keywords, creating your own exception, multiple catch statements.	10
Unit-5	Java Input Output: Java IO package, File, Class Byte/Character Stream, Buffered reader / writer, File reader / writer; Multithreading: Multithreading concept, Java thread model, Main thread, Creating a thread, Creating multiple threads, thread priorities.	9

Text Books:

1. Balaguruswamy, Programming with JAVA A primer, 4th Edition, TATAMcGraw-Hill
2. Java Complete Reference, Fourth edition,TMH,
3. Java 2 - The Complete Reference – McGraw Hillpublication.
4. Java - The Complete Reference, 7th Edition, By Herbert Schildt– McGraw Hill publication.

References:

1. Object Oriented Programming with Java : Somashekara, M.T., Guru, D.S., Manjunatha K.S
2. Cay S Horstmann, Fary Cornell, Core Java 2, Volume – I, Sun Microsystems Press
3. Core Java Volume I – Fundamentals, By Cay S. Horstmann, PrenticeHall

Year	I	Course Code: SEPBCAP2.6	Credits	2
Semester	II	Course Title: Data Structures Lab	Hours	48
Course Pre-requisite, if any				
Formative Assessment Marks: 10		Summative Assessment Marks: 40	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course the student should be able to: o Design the algorithm to analyse the problem o Differentiate between iteration and recursion and apply recursive definition for problem solving o Apply various types of searching and sorting techniques on arrays o Evaluate the applications of stacks, queues, linked lists and tree		

Lab Assignments

Part A:

1. Program to implement dynamic array, find smallest and largest element of the array.
2. Program to find GCD using recursive function
3. Program to generate binomial coefficient using recursive function.
4. Program to generate n Fibonacci numbers using recursive function.
5. Program to implement Towers of Hanoi using recursion.
6. Write a program that takes a file as an argument and counts the total number of lines. Lines are defined as ending with a newline character.
7. Program to search an element using linear search technique.
8. Program to search an element using binary search technique.
9. Program to create a text file and count the characters present in it.
10. Program to copy the contents of one text file to another text file.

Part B:

1. Program to sort the given list using selection sort technique.
2. Program to sort the given list using bubble sort technique.
3. Program to sort the given list using insertion sort technique.
4. Program to sort the given list using quick sort technique.
5. Program to sort the given list using merge sort technique.
6. Program to implement Stack.
7. Program to convert an infix expression to postfix.
8. Program to implement simple queue.
9. Program to implement linear linked list.
10. Program to implement a binary tree and traverse in pre-order

<table>
<tr><td>Year</td><td>I</td><td>Course Code: SEPBCAP2.7</td><td>Credits</td><td>2</td></tr>
<tr><td>Semester</td><td>II</td><td>Course Title:
Numerical and Statistical Methods Lab</td><td>Hours</td><td>48</td></tr>
<tr><td colspan="2">Course Pre-requisite, if any</td><td colspan="3">NA</td></tr>
<tr><td colspan="2">Formative Assessment Marks: 10</td><td>Summative Assessment Marks: 40</td><td colspan="2">Duration of ESA: 03 hrs.</td></tr>
<tr><td colspan="2">Course Outcomes</td><td colspan="3">At the end of the course the student should be able to:
o Learn how to obtain numerical solution of nonlinear equations using Bisection, Newton – Raphson and fixed-point iteration methods.
o Solve system of linear equations numerically using direct and iterative methods
o Develop a framework for estimating and predicting the different sample of data for handling the uncertainties.
o Understand the various approaches dealing the data using theory of probability</td></tr>
</table>

Lab Assignments

(Part A must be implemented using C Language and Part B using SPSS)

Part A

1. Write a program to check whether the given matrix is singular or not.
2. Write a program to find roots of an equation f(x)=0 using Bisection method.
3. Write a program to find roots of an equation f(x)=0 using Regula-Falsi method.
4. Write a program to find roots of an equation f(x)=0 using Newton-Rephson method.
5. Write a program to solve the system equation Ax=b using Gauss Elimination method.
6. Writea program to solve the system of equation Ax=b using Gauss Seidel method.

Part B:

1. Open a new dataset in SPSS
   - Create a nominal variable called cat_dog that has a width of 3 with 0 decimal places. The label should be "Do you like cats or dogs better?" The values should be 1 for cats and 2 for dogs (or vice-versa). Do not worry about missing data codes.
   - Create a scale variable called neatness that has a width of 8 with 3 decimal places. The label should be "Eric Cartman's Neatness Scale(higher=neater)". There will be no value labels.
   - Enter data for the following cases
     - ➢ Case 1 prefers cats and has a neatness of 4
     - ➢ Case 2 prefers dogs and has a neatness of 3
     - ➢ Case 3 prefers dogs and has a neatness of 7
     - ➢ Case 4 prefers dogs and has a neatness of 2
     - ➢ Case 5 prefers cats and has a neatness of 5
     - ➢ Case 6 prefers cats and has a neatness of 1
     - ➢ Case 7 prefers cats and has a neatness of 3
     - ➢ Case 8 prefers dogs and has a neatness of 6
   - Change the neatness of the second case from 3 to 6, like you would if you discovered a data entry error.

2. Create a dataset in SPSS for the following data:

[

Group	Gender	Hw1	Hw2	Hw3
expt	Male	92	84	93
expt	Female	77	84	85
expt	Male	87	86	81
expt	Female	89	90	93
expt	Male	64	73	78
control	Female	81	84	93
control	Male	83	90	91
control	Female	84	88	86
control	Male	82	80	78
control	Female	96	91	88

- Using the Frequencies option, find the mean, median, mode, quartiles, 95th percentile, variance, standard deviation, minimum, and maximum of Hw1, Hw2, and Hw3.
- Using the Descriptive option, find the means and standard deviations of Hw1,Hw2 and Hw3.
- Using the Compare Means - Means procedure, find the mean son Hw1,Hw2 and Hw3 for everyone for the experimental group, for the control group, for men, for women, and for all combinations of gender and group.

3. A researcher has created a data table showing the anthropometrical measurements of tribal subjects under each of the four social categories, namely GM, OBC, SC and ST as shown in table.

GM			
Gender	HT	WT	Biceps
1	137.8	30.5	5.50
2	130.2	29.5	5.65
2	135.6	29.0	5.15
2	137.8	30.0	5.25
1	131.5	30.5	4.95
1	132.8	31.5	5.65
1	139.8	30.5	5.50
1	136.7	30.0	5.65
1	138.6	30.5	5.15
1	139.5	30.5	5.25

SC			
Gender	HT	WT	Biceps
2	132.4	25.0	4.37
1	133.5	24.5	4.95
1	130.6	25.5	4.65
1	132.5	26.5	4.45
1	130.6	26.0	6.48
2	132.4	25.5	5.01
1	130.5	25.0	4.37
1	132.4	24.5	4.95
2	133.5	25.5	4.65
2	130.6	26.5	4.45

OBC			
Gender	HT	WT	Biceps
1	124.4	23.5	4.61
2	125.5	23.0	4.52
1	126.3	24.0	4.45
2	128.0	23.5	4.39
1	129.0	25.0	4.37
2	130.0	22.0	4.69
1	129.5	23.5	4.61
1	130.0	23.0	4.52
2	126.0	24.0	4.45
2	128.5	23.5	4.39

ST			
Gender	HT	WT	Biceps
1	124.5	20.5	3.54
1	125.8	21.0	3.55
1	123.5	20.5	3.95
1	124.8	22.0	4.05
1	122.5	21.5	3.55
1	122.8	22.0	3.54
1	122.5	22.5	3.55
1	121.5	21.5	3.95
1	124.5	20.5	4.05
2	125.8	21.0	3.55

- Create a data file in SPSS (The Data in SPSS has to be entered with category1=GM,2=OBC,3=SC and 4=ST. The codes for Gender are 1=Male and 2= Female).

- Generate central tendency and measures of dispersion output using the descriptives command in SPSS for the variables Height, Weight and Biceps.
- Generate two-way cross table Gender versus Category.

4. The marks obtained by 50 students of a class in mathematics are given below.

32,42,41,51,41,30,39,18,48,53,54,32,31,46,15,37,32,56,42,48,38,26,50,40,38,42,35,22,62,51,44, 21,45,31,37,41,44,18,37,47,38,41,30,52,52,60,42,38,38,34.

- Create a data file in SPSS.
- Generate a frequency table.
- Draw the Histogram.
- Generate central tendency output using the frequencies command in SPSS.
- Generate central tendency output using the descriptives command in SPSS.
- Generate central tendency output using the explore command in SPSS.

5. The number of blood donations in the years 1995 and 2000 in various blood groups are as follows

Year	O	A	B	AB
1995	1154	526	775	155
2000	700	1125	1280	560

Create a data file in SPSS and hence represent the data by multiple bar diagram.

6. Fitting the Binomial & Poisson probability distribution for the following data using SPSS

x	0	1	2	3
f	36	40	22	2

Year	I	Course Code: SEPBCAP2.8	Credits	2
Semester	II	Course Title: Object Oriented Programming using Java - Lab	Hours	48
Course Pre-requisite, if any		Programming Language concepts		
Formative Assessment Marks: 10		Summative Assessment Marks: 40	Duration of ESA: 03 hrs.	
Course Outcomes		At the end of the course, the students should be able to: o Write, compile, and execute Java programs that may include basic data types and control flow constructs and how type casting is done. o Identify classes, objects, members of a class and relationships among them needed for a specific problem and demonstrate the concepts of polymorphism and inheritance o Demonstrate programs based on arrays and strings. o Demonstrate programs based on interfaces and threads		

Lab Assignments(to be implemented using Netbeans/eclipse or any other Open Source IDE)

Part A:

1. Program to simple java application, to print the message, “Welcome to java”
2. Program to display the month of a year. Months of the year
   should be held in an array.
3. Program to assign two integer values to X and Y. Using the „if" statement the output of the program should display a message whether X is greater than Y.
4. Program to find the area of rectangle.
5. Write a Java program to find factorial of a number reading input as command line argument.
6. Write a Java program to find number characters entered by user reading input as command line argument.
7. Write a Java program that creates an object and initializes its data members using constructor. Use constructor overloading concept.
8. Write a Java program to demonstrate method overloading.
9. Write a Java program to demonstrate static variables.
10. Write a Java program to demonstrate static methods

Part B:

1. Program to demonstrate single inheritance
2. Program to demonstrate multilevel inheritance. Show the usage of super().
3. Write a program to demonstrate use of user defined package by importing the package and access the member variable of classes contained in the package.
4. Write a java program to demonstrate at least 5 string methods from (String class).
5. Write a java program to demonstrate string methods (String Buffer class).
6. Write a program to demonstrate use of implementing interfaces.
7. Write a java program to implement exception handling using multiple catch statements. Also include code to identify the significance of finally block in handling exceptions.
8. Illustrate creation of thread by Extending Thread class.
9. Illustrate creation of thread by Implementing Runnable interface
10. Write a Java Program to implement inheritance and demonstrate use of method overriding.

Scheme of Evaluation
Semester End Exam Question Paper Pattern

Duration of the examination: 3hour Max. Marks: 80

Section A

Answer any TEN Questions from the following, each carries 2 marks: [10X2=20]

1. --
2. --
3. --
4. --
5. --
6. --
7. --
8. --
9. --
10. --
11. --
12. --

Section B

Answer any FOUR from the following questions each carries 5 marks. [4X5=20]

13. --
14. --
15. --
16. --
17. --

Section C

Answer any FOUR from the following questions each carries 10 marks. [4X10=40]
(The Question may consists two sub-questions)

18. --
19. --
20. --
21. --
22. --

CIA for Practical	
Assessment Type	Marks
Test1	10
Total	**10 Marks**

Instructions:

1. Based on Two practical Tests 10 marks of practical shall be awarded.

Note: Guidelines given by the university from time-to-time shall be followed for IA.

Semester End Examination Scheme of Evaluation for Lab Examination

Assessment Criteria	Marks
Writing of 2 Programs (Each from Part A & Part B)	15
Execution (Includes program code modification and execution result)	15
Journal	05
Viva Voice	05
Total	**40 Marks**

Instructions:

1. Certified Journal is mandatory for appearing in the Lab examination.
2. Students shall be given two programming assignments taking into consideration of duration of the time allotted to students for writing, typing and executing the programs.

CIA for Theory	
Assessment Type	Marks
Test 1 / Seminar / Activity	10
Test 2 / Seminar / Activity	10
Total	**20 Marks**