**Model Curriculum Structures for Bachelor of Science Programme
with Computer Science as a Major Course Model Syllabus
for I to IV Semesters and Elective Courses in Computer Science
As per**

Modified CBCS Scheme

2024-25 & 2025-26 on words

{BoS Meeting Held on 26 - 07 - 2025}

DEPARTMENT OF COMPUTER SCIENCE

The objectives of the B.Sc. - Computer Science Program

1. The primary objective of this program is to provide a foundation of computing

2. It helps students analyze the requirements for system programming and exposes

3. This programme provides students with options to specialize in various software

4. To produce outstanding Computer Scientists who can apply the theoretical knowledge into practice in the real world and develop standalone live projects themselves.

5. To provide opportunity for the study of modern methods of information processing and its applications.

6. To develop among students programming techniques and the problem- solving skills through programming

7. To prepare students who wish to go on to further studies in computer science and related subjects.

8. To acquaint students working effectively with a range of current, standard, Office Productivity software applications.

Program Outcomes: BSc. (Computer Science) Degree

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

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

3. **Design and Development of Solutions:** Ability to design and development of algorithmic solutions to real-world problems.

4. **Programming a computer:** Exhibiting strong skills required to program a computer for various issues and problems of day-to-day scientific applications.

5. **Application Systems Knowledge**: Possessing a minimum knowledge to practice existing computer application software.

6. **Communication:** Must have reasonably good communication knowledge both orally and in writing.

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

8. **Lifelong Learning:** Should become an independent learner. So, learn to learn ability.

9. **Motivation to take up Higher Studies:** Inspiration to continue educations towards advanced studies on Computer Science.

Additional Program Outcomes for B. Sc. (Hons) in Computer Science

The four years Bachelors in Computer Science (Hons) program enables students to attain the following additional attributes besides the afore-mentioned attributes:

1. Apply standard Software Engineering practices and strategies in real-time software project development

2. Design and develop computer programs/computer-based systems in the areas related to AI, algorithms, networking, web design, cloud computing, IoT and data analytics.

3. Acquaint with the contemporary trends in industrial/research settings and thereby innovate novel solutions to existing problems

4. The ability to apply the knowledge and understanding noted above to the analysis of a given information handling problem.

5. The ability to work independently on a substantial software project and as an effective team member.

Program Structures for the Undergraduate Program Bachelor of Science in Physics, Mathematics, Computer Science with Practical						
Semester	**Coerces**	**Type**	**Credits**	**L: T: P**	**Marks**	
I	Major 1to 3	Theory	9	3: 0: 0	**300**	
	Major 1 to 3	Practical	6	0: 0: 2	**150**	
	Language 1	Theory	3	3: 0: 0	**100**	
	Language 2	Theory	3	3: 0: 0	**100**	
	Constitutional Values	Theory	2	2: 0: 0	**50**	
II	Major 1to 3	Theory	9	3: 0: 0	**300**	
	Major 1 to 3	Practical	6	0: 0: 2	**150**	
	Language 1	Theory	3	3: 0: 0	**100**	
	Language 2	Theory	3	3: 0: 0	**100**	
	Constitutional Values	Theory	2	2: 0: 0	**50**	
III	Major 1to 3	Theory	9	3: 0: 0	**300**	
	Major 1 to 3	Practical	6	0: 0: 2	**150**	
	Language 1	Theory	3	3: 0: 0	**100**	
	Language 2	Theory	3	3: 0: 0	**100**	
	Elective 1	Theory	3	3: 0: 0	**100**	
	Elective 2	Theory	3	3: 0: 0	**100**	
IV	Major 1to 3	Theory	9	3: 0: 0	**300**	
	Major 1 to 3	Practical	6	0: 0: 2	**150**	
	Language 1	Theory	3	3: 0: 0	**100**	
	Language 2	Theory	3	3: 0: 0	**100**	
	Elective 3	Theory	3	3: 0: 0	**100**	
	Compulsory Paper	Practical/Skill	2	2: 0: 0	**50**	

Modified CBCS Syllabus – BSc Computer Science, for 2024-25 onwards

Year	Sem	Course Code	Title	Hours / Week			Credits			Maximum Marks						Exam Duration	Total Marks
										Th. IA		Pr. IA		Exam			
				L	T	P	L	T	P	C1	C2	C1	C2	Th.	Pr.		
I	I	GCS 101 (Theory)	Computer Concepts and Problem-Solving using C++	3	0	0	3	0	0	10	10	-	-	80	-	3 Hours	100
		GCS 102 (Practical)	Computer Basics and Programming in C++	0	0	4	0	0	2	-	-	05	05	-	40	3 Hours	50
	II	GCS 201 (Theory)	Data Structures	3	0	0	3	0	0	10	10	-	-	80	-	3 Hours	100
		GCS 202 (Practical)	Data Structures using C++	0	0	4	0	0	2	-	-	05	05	-	40	3 Hours	50
II	III	GCS 301 (Theory)	Database Management System	3	0	0	3	0	0	10	10	-	-	80	-	3 Hours	100
		GCS 302 (Practical)	DBMS Lab	0	0	4	0	0	2	-	-	05	05	-	40	3 Hours	50
		GCS 303 (Theory)	*Elective 1: Software Engineering	3	0	0	3	0	0	10	10	-	-	80	-	3 Hours	100
		GCS 304 (Theory)	*Elective 2: E-Commerce & E-2Governance	3	0	0	3	0	0	10	10	-	-	80	-	3 Hours	100
	IV	GCS 401 (Theory)	Object Oriented Programming with Java	3	0	0	3	0	0	10	10	-	-	80	-	3 Hours	100
		GCS 402 (Practical)	Java Lab	0	0	4	0	0	2	-	-	05	05	-	40	3 Hours	50
		GCS 403 (Theory)	*Elective 3: Software Testing	3	0	0	3	0	0	10	10	-	-	80	-	3 Hours	100
		GCS 404 (Practical)	Compulsory Paper: Digital Marketing / Other	2	0	0	2	0	0	05	05	-	-	40	-	2:30 Hours	50
		*Elective:	Any one Elective Course offering for Computer Science in 3rd / 4th Semester														

Modified CBCS Course Content for BSc. Computer Science, Semesters I and II

Semester: I

Course Code: GCS101	**Course Title:** Computer Concepts and Problem Solving Using C++
Course Credits: 03 (3-0-0)	**Hours/Week:** 03
Total Contact Hours: 44	**Formative Assessment Marks:** 20
Exam Marks: 80	**Exam Duration:** 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

CO1:   Understand the basics of computers.

CO2:   Understand problem-solving strategies and techniques.

CO3:   Describe the Object-Oriented Programming principles and concepts.

CO4:   Understanding of the syntax and semantics of C++.

Unit – 1 Course Contents	Hours
Introduction To Computers: Definition And Characteristics of Computers, Brief History of Computers, Classification of Computers Based on Size and Processing Ability. Applications Of Computers. **Computer Architecture:** CPU, ALU, Control Unit, Registers, Cache Memory, RAM, ROM, Input/Output Components, Buses, Ports. Hardware And Software: System Software [Operating System, Interpreters], Application Software. Languages Of Computer, ASCII And EBCDIC, Computer Threats and Safety Measurements.	11
Unit-2	
Problem-Solving Techniques: Problem, Definition, Analysis, Design Tools [Algorithm & Flow Chart], Coding, Testing, Maintenance. Basics Of Algorithm Hours Analysis: Time Complexity, Space Complexity, Asymptotic Notations. **Basic Programming Concepts:** Tokens Of Programming Language, Identifiers, Constants, Variables, Data Types, And Operators. Introduction To Programming Concepts. Comparison Of POP And OOP.	11
Unit-3	
Introduction To C++: Overview Of Programming Languages, History and Features of C++, Structure of C++ Program, Data Types in C++. Control Structures: Decision Making: If, If-Else, Nested If, Switch Statement. Looping: While, Do-While, For. Unconditional Statements: Break And Continue Statements. **Access Specifiers:** Public, Private, Protected. Pointers In C++.	11

Unit-4	
Objects, Classes: Base and Derived Classes, Data Encapsulation, Data Abstraction, Friend Function, Inheritance: Single, Multiple, Multilevel, Hierarchical.	**11**
Polymorphism: Function Overloading, Operator Overloading, Virtual Functions.	
Constructors And Destructors: Exception Handling.	

Reference:

1. "Programming: Principles and Practice Using C++" by Bjarne Stroustrup.
2. "C++ Primer" by Stanley B. Lippman, Josée Lajoie, and Barbara E. Moo.
3. "Object-Oriented Programming with C++" by E. Balagurusamy
4. "Problem Solving and Object-Oriented Programming with C++" by Tanenbaum and Mukherjee:

Course Code: GCS102	Course Title: Computer Basics and Programming in C++
Course Credits: 02 (0-0-2)	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

CO1: Illustrate the hardware components of a computer.

CO2: Use Open-Source Office tools.

CO3: Demonstrate simple programming skills through the C++ programming language.

Laboratory Program List

PART -A

1. Demonstration of Desktop computer Hardware components.
2. Demonstration of Word Processor software [Open-Office].
3. Demonstration of Spreadsheet software [Open-Office].
4. Demonstration of Presentation software [Open-Office].
5. Write a C++ program to print the user's name.
6. Write a C++ program to swap 2 numbers.
7. Write a C++ Program to Find the Largest Among 3 Numbers
8. Write a C++ program to perform arithmetic operations.

PART-B

1. Write a C++ Program to Print the Multiplication Table of a Number
2. Write a C++ Program to Reverse a Number
3. Write a C++ Program to Check Whether a Number is a Palindrome or Not.
4. Write a C++ Program to Create a Class and an Object
5. Write a C++ program for single inheritance.
6. Write a C++ program to demonstrate multi-level inheritance.
7. Write a C++ program to demonstrate operator overloading.
8. Write a C++ Program by using Pointers

Semester II

Course Code: GCS201	**Course Title:** Data Structures
Course Credits: 03 (3-0-0)	**Hours/Week:** 03
Total Contact Hours: 44	**Formative Assessment Marks:** 20
Exam Marks: 80	**Exam Duration:** 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

CO1:   Understand the basics of Data Structures.
CO2:   Identify the appropriate data structures and algorithms for solving real-world problems.
CO3:   Understand the practical applications of Trees and Graphs.

Unit – 1	Course Contents	Hours
Introduction: Data Structure Definition, Basic Terminology and Concepts, Importance of Data Structures In Programming. Classification of Data Structures. Primitive Data Structures, Non-Primitive Data Structures. **Arrays:** Declaration, Initialization, Accessing Elements, Multidimensional Arrays. C++ Strings, String Handling Functions, Applications of Arrays and Strings. **Stack:** Definition, Memory Representation, Algorithms for Stack Operations (Push, Pop), Applications of Stack.		11
Unit-2		
Queue: Definition, Memory Representation, Linear Queue, Circular Queue, Enqueue, Dequeue. Applications Of Queue. **Linked Lists**: Definition, Types. **Singly Linked List:** Implementation, Insertion [At the Beginning], Deletion [At the End]. **Doubly Linked List:** Memory Representation of Singly Linked List and Doubly Linked List. Applications of Linked List.		11
Unit-3		
Tree: Definition, Memory Representation Using Array and Linked List. **Binary Tree:** Definition, Traversal Algorithms [Pre-Order, In-Order, Post-Order], Construction of Tree from In-Order and Pre-Order, In-Order and Post-Order. Binary Search Trees: Insertion of a Node, Deletion of A Node. **Advanced Tree Structures AVL And B-Trees:** Definition and Applications.		11

Unit-4	
Graph: Definition, Memory Representation of Graph. Adjacency Matrix, Adjacency List. Graph Traversal Algorithms: Breadth-First Search (BFS), Depth-First Search (DFS). **Sorting Techniques:** Bubble Sort, Selection Sort [Algorithm, Time & Space Complexity]. **Searching Techniques:** Linear And Binary Search Sort [Algorithm, Time & Space Complexity]. **Heap:** Heap Operations and Applications.	**11**

Reference Books:

1. Data Structures Through C++ (4th Edition) Yashvant Kanetkar.
2. Data Structures and Algorithm Analysis in C++" by Mark Allen Weiss.
3. Data Structure and Algorithms using C++ by Sachi Nandan Mohanty, Pabitra Kumar Tripathy.
4. Data Structures and Algorithms in C++, Second Edition by Adam Drozdek.

Course Code: GCS202	**Course Title:** Data Structures using C++
Course Credits: 02 (0-0-2)	**Hours/Week:** 04
Total Contact Hours: 60	**Formative Assessment Marks:** 10
Exam Marks: 40	**Exam Duration:** 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

CO1: Implement data structures using C++.

CO2: Demonstrate searching and sorting techniques using ++.

CO3: Demonstrate simple programming skills through the C++ programming language.

Laboratory Program List

Part A:

1. C++ program to find GCD of two numbers.
2. C++ program to implement Tower of Hanoi.
3. C++ program to print Fibonacci series.
4. C++ program to find largest and smallest element in an array.
5. C++ program to perform stack operations.
6. C++ program to perform Linear queue operations
7. C++ program to insert a node at the beginning of a singly linked list.
8. C++ program to delete a node at the end of a singly linked list.

Part B:

1. C++ program to construct a binary search tree
2. C++ program for Binary Tree traversal.
3. C++ program to implement DFS
4. C++ program to implement BFS
5. C++ Program to Sort an Array (Selection Sort)
6. C++ Program to Sort an Array (Bubble Sort)
7. C++ Program to perform Linear Search of an Element in an Array.
8. C++ Program to perform Binary Search of an Element in an Array.

Semester: III

Course Code: GCS 301	**Course Title: Database Management System**
Course Credits: 03 (3-0-0)	**Hours/Week:** 03
Total Contact Hours: 44	**Formative Assessment Marks:** 20
Exam Marks: 80	**Exam Duration:** 03

Course Outcomes (COs):

CO1: Understand basic concepts of databases, data models, and ER diagrams.
CO2: Apply relational algebra and SQL to query and manipulate data.
CO3: Analyze database normalization to improve design and remove anomalies.
CO4: Evaluate transaction management and database security features.

Content	Hours
Unit-1	
Introduction to Databases: Definition of Data, Database, and DBMS, Overview of Database, Applications, Advantages and Disadvantages of DBMS, Roles of Database Users and Administrators. **Data Models:** Introduction to Data Models, Types of Data Models (Hierarchical, Network, Relational, Object-oriented), Importance of Data Models in DBMS. **Database Design**: Keys: Primary Key, Candidate Key, Super Key, Foreign Key, Composite Key, Alternate Key, Unique Key, Surrogate Key, Constraints in a table: Primary Key, Foreign Key, Unique Key, NOT NULL, CHECK, Entity-Relationship (ER) Model, Entities and Entity Sets, Attributes and Relationships, ER Diagrams, Key Constraints and Weak Entity Sets, Extended ER Features, Introduction to the Relational Model and Relational Schema.	11
Unit-2	
Relational Algebra: Introduction to Relational Algebra, **Operations:** Selection, Projection, Set Operations, Join Operations, Division. **Structured Query Language (SQL):** SQL Basics: DDL and DML, Aggregate Functions (Min (), Max(), Sum(), Avg(), Count ()), Logical operators (AND, OR, NOT), Predicates (Like, Between, Alias, Distinct), Clauses (Group By, Having, Order by, top/limit).	11
Unit-3	
SQL Joins and Views: Inner Join, Natural Join, Full Outer Join, Left Outer Join, right outer Join, Equi Join, Definition of View, creating a View, Managing Views (Listing, Updating, Deleting). **Normalization**: Anomalies in relational database design. Functional dependencies - Axioms. Decomposition, Transitive Dependency. **Data Normalizations:** First normal form, Second normal form, Third normal form. Boyce-Codd normal form.	11
Unit-4	
Query Processing Transaction Management: Introduction Transaction Processing, Single user & multiuser systems. **Transactions**: read & write operations. **Need of concurrency control:** The lost update problem, Dirty read problem. Types of failures. Transaction states. Desirable properties (ACID properties) of Transactions. Storage of Database, File Operations, and Database Security.	11

Reference Books:

1. Fundamentals of Database Systems, Ramez Elamassri, Shankant B. Navathe, 7thEdition, Pearson.

Reference Books:

1. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010.
2. Introduction to Database System, C J Date, Pearson, 1999.
3. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill,
4. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw Hill.

Semester: III

Course Code: GCS 302	**Course Title:** DBMS Lab
Course Credits: 02 (0-0-2)	**Hours/Week:** 04
Total Contact Hours: 60	**Formative Assessment Marks:** 10
Exam Marks: 40	**Exam Duration:** 03

Course Outcomes (COs):
CO1: Execute single-line SQL queries and apply group functions effectively.
CO2: Perform database operations using DDL, DML, DCL, and TCL commands.
CO3: Implement advanced SQL concepts like nested queries and join operations.
CO4: Create views and apply table-level locking mechanisms for data control.

Laboratory Program List

Part A
Activity 1: Database: Student (DDL, DML Statements)

Table: Student

Name	Reg. No	Class	Major
Smith	17	1	CS
Brown	8	2	CS

Table: Course

Course Name	Course Number	Credit Hours	Department
Introduction to Computer Science	CS1310	4	CS
Data Structure	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database Management System	CS3380	3	CS

Table: Section

Section Identifier	Course Number	Year	Instructor
85	MATH2410	98	King
92	CS1310	98	Andreson
102	CS3320	99	Knuth
112	MATH2410	99	Chang
119	CS1310	99	Andreson
135	CS3380	99	Stone

Table: Grade_Report

Reg. No	Section Identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

Queries

1. Create Table Using create statement.
2. Insert rows into individual Tables using insert statement.
3. Alter table section add new field section and update the records
4. Delete brown's grade report.
5. Drop the table section.

Activity 2: (Select clause, Arithmetic Operators)

Database: Employee

Create the following tables and insert tuples with suitable constraints.

Table: EMPLOYEE

EMPID	FIRSTNAME	LASTNAME	Hire Date	ADDRESS	CITY
1001	George	Smith	11-May-06	83 First Street	Paris
1002	Mary	Jones	25-Feb-08	842 Vine Ave	Losantiville
1012	Sam	Tones	12-Sep-05	33 Elm St.	Paris
1015	Peter	Thompson	19-Dec-06	11 Red Road	Paris
1016	Sarath	Sharma	22-Aug-07	440 MG Road	New Delhi
1020	Monika	Gupta	07-Jun-08	9 Bandra	Mumbai

Table: EMPSALARY

EMPID	SALARY	BENEFITS	DESIGNATION
1001	10000	3000	Manager
002	8000	1200	Salesman
1012	20000	5000	Director
1015	6500	1300	Clerk
1016	6000	1000	Clerk
1020	8000	1200	Salesman

Queries

1. Display FIRSTNAME, LASTNAME, ADDRESS, and CITY of all employees living in PARIS
2. Display the content of the employee table in descending order of FIRSTNAME
3. Select FIRSTNAME and SALARY of salesmen

4. Display the FIRSTNAME, LASTNAME, and TOTAL SALARY of all employees where TOTAL SALARY = SALARY + BENEFITS
5. Count the number of distinct DESIGNATIONs from EMPSALARY
6. List the employees whose names have exactly 6 characters
7. Add a new column PHONE_NO to the EMPLOYEE table and update the records
8. List employee names who have joined before 15-Jun-08 and after 16-Jun-07
9. Generate salary slip with Name, Salary, Benefits, HRA=50%, DA=30%, PF=12%, and calculate gross salary. Order by gross salary in descending order.

Activity 3: (Logical, Relational Operators)
Database: Library

Create the following tables and insert tuples with suitable constraints. **Table:**
Books

Book_Id	Book_Name	Author_Name	Publishers	Price	Type	Quantity
C0001	The Klone and I	Lata Kappor	EPP	355	Novel	5
F0001	The Tears	William Hopkins	First Publ	650	Fiction	20
T0001	My First C++	Brain & Brooke	First Publ	350	Text	10
T0002	C++ Brainwork's	A.W. Rossaine	TDH	350	Text	15
F0002	Thunderbolts	Ana Roberts	First Publ	750	Fiction	50

Table: Issued

Book_Id	Quantity_Issued
T0001	4
C0001	5
F0001	2
T0002	5
F0002	8

Queries
1. To show the Book name, Author name and price of books of First Publ. publisher.
2. Display Book id, Book name and publisher of books having quantity more than 8 and price less than 500.
3. Select Book id, book name, author name of books which is published by other than ERP publishers and price between 300 to 700.
4. Generate a Bill with Book_id, Book_name, Publisher, Price, Quantity, 4% of VAT "Total".
5. Display book details with book id"s C0001, F0001, T0002, F0002 (Hint: use IN operator).
6. Display Book list other than, type Novel and Fiction.
7. Display book details with author name starts with letter "A".
8. Display book details with author name starts with letter "T" and ends with "S".
9. Select Book_Id, Book_Name, Author Name , Quantity Issued where Books.Books_Id = Issued.Book_Id.
10. List the book_name, Author_name, Price. In ascending order of Book_name and then on descending order of price.

Activity 4: (Date Functions)

Database: Lab

Create the following table and insert tuples with suitable constraints.

Table: Equipment Details

No.	Item Name	Cost Per Item	Quantity	Date of Purchase	Warranty	Operational
1	Computer	30000	9	21/5/07	2	7
2	Printer	5000	3	21/5/06	4	2
3	Scanner	8000	1	29/8/08	3	1
4	Camera	7000	2	13/6/05	1	2
5	UPS	15000	5	21/5/08	1	4
6	Hub	8000	1	31/10/08	2	1
7	Plotter	25000	2	11/1/09	2	2

Queries

1. To select the ItemName purchase before 31/10/07.
2. Extend the warranty of each item by 6 months.
3. Display ItemName, Dateof purchase and number of months between purchase date and present date.
4. To list the ItemName in ascending order of the date of purchase where quantity is more than 3.
5. To count the number, average of costperitem of items purchased before 1/1/08.
6. To display the minimum warranty, maximum warranty period.
7. To Display the day of the date, month, and year of purchase in characters.
8. To round off the warranty period to a month and year format.
9. To display the next Sunday from the date "07-JUN-96".
10. To list the ItemName, which are within the warranty period till the present date.

Part B:

Activity 5: (Numeric, character functions).

Use Functions for the following.

1. Find the mod of 165,16.
2. Find Square Root of 5000.
3. Truncate the value 128.3285 to 2 and -1 decimal places
4. Round the value 92.7683 to 2 and -1 decimal places.
5. Convert the string 'Department' to uppercase and lowercase.
6. Display your address convert the first character of each word to uppercase and rest are in lowercase.
7. Combine your first name and last name under the title Full name.
8. A) Take a string length maximum of 15 displays your name to the left. The remaining space should be filled with '*'.
9. Take a string length maximum of 20 displays your name to the right. The remaining space should be filled with '#'.
10. Find the length of the string 'JSS College, Mysore'.
11. Display substring 'BASE' from 'DATABASE'.
12. Display the position of the first occurrence of character 'o' in Position and Length.
13. Replace string Database with Data type.
14. Display the ASCII value of ' ' (Space).
15. Display the Character equivalent of 42.

Activity 6: Database: subject

Create the following table and insert tuples with suitable constraints.

Table: Physics

Regno	Name	Year	Combination
AJ00325	Ashwin	First	PCM
AJ00225	Swaroop	Second	PMCs
AJ00385	Sarika	Third	PME
AJ00388	Hamsa	First	PMCs

Table: Computer Science

Regno	Name	Year	Combination
AJ00225	Swaroop	Second	PMCs
AJ00296	Tejas	Second	BCA
AJ00112	Geetha	First	BCA
AJ00388	Hamsa	First	PMCs

Queries

1. Select all students from Physics and Computer Science.
2. Select student common in Physics and Computer Science.
3. Display all student details who are studying in the second year.
4. Display students who are studying both Physics and Computer Science in the second year.
5. Display the students studying only Physics.
6. Display the students studying only Computer Science.
7. Select all students having PMCs combination.
8. Select all students having BCA combination.
9. Select all students studying in the third year.
10. Rename table Computer Science to CS.

Activity 7: (views)
Database: Railway Reservation System.
Create the following table and insert tuples with suitable constraints.

Table: Train Details

Train_No	Train_Name	Start_Place	Destination
RJD16	Rajdhani Express	Bangalore	Mumbai
UDE04	Udhyan Express	Chennai	Hyderabad
KKE55	Karnataka Express	Bangalore	Chennai
CSE3	Shivaji Express	Coimbatore	Bangalore
JNS8	Janashatabdi	Bangalore	Salem

Table: Availability

Train_No	Class	Start_Place	Destination	No_of_seats
RJD16	Sleeper Class	Bangalore	Mumbai	15
UDE04	First Class	Chennai	Hyderabad	22
KKE55	First Class AC	Bangalore	Chennai	15
CSE3	Second Class	Coimbatore	Bangalore	8
JNS8	Sleeper Class	Bangalore	Salem	18

Queries

1. Create a view **sleeper** to display train number, start place, destination, which have sleeper class and perform the following: a. Insert a new record.
   b. Update destination='Mangalore' where train no='RJD16'.
   c. Delete a record which has train no = ' KKE55'
2. Create a view **detail** to display the train number, train name, and class.
3. Create a view **total seat** to display train number, start place, use COUNT function on No_of_seats, group by start place and perform the following: a. Insert a new record.
   b. Update start place='Hubli' where train no='JNS8'.
   c. Delete the last row of the view.
4. Rename view **sleeper** to **class**.
5. Delete view **details**.

Activity 8: (group by, having clause)
Database: Bank system
Create the following table and insert tuples with suitable constraints.
Table: Account

Account_No	Cust_Name	Branch_ID
AE0012856	Reena	SB002
AE1185698	Akhil	SB001
AE1203996	Daniel	SB004
AE1225889	Roy	SB002
AE8532166	Sowparnika	SB003
AE8552266	Anil	SB003
AE1003996	Saathwik	SB004
AE1100996	Swarna	SB002

Table: Branch

Branch_ID	Branch_Name	Branch_City
SB001	Malleshwaram	Bangalore
SB002	MG Road	Bangalore
SB003	MG Road	Mysore
SB004	Jayanagar	Mysore

Table: Depositor

Account_No	Branch_ID	Balance
AE0012856	SB002	12000
AE1203996	SB004	58900
AE8532166	SB003	40000
AE1225889	SB002	150000

Table: Loan

Account_No	Branch_ID	Balance
AE1185698	SB001	102000
AE8552266	SB003	40000
AE1003996	SB004	15000
AE1100996	SB002	100000

Queries

1. Display the total number of accounts present in each branch.
2. Display the total loan amount in each branch.
3. Display the total deposited amount in each branch in descending order.
4. Display the maximum and minimum loan amount present in each city.
5. Display the average amount deposited in each branch for each city.
6. Display the maximum loan amount in each branch where the balance is more than 25000.
7. Display the total number of accounts present in each city.
8. Display all customer details in ascending order of branch ID.
9. Update the balance to 26000 where Account_No = AE1003996.
10. Display customer names with their branch name.

Evaluation Scheme for Lab Examination

Assessment Criteria Marks		Marks
Writing	One Program from Part A	15
	One Program from Part B	15
Execution	Any one of the Written Programs	5
Viva Voce based on Database Management System		5
Total		**40**

Semester: III

Course Code: GCS 303 **(Elective 1)**	**Course Title: Software Engineering**
Course Credits: 03(3-0-0)	**Hours/Week:** 03
Total Contact Hours: 44 Hours	**Formative Assessment Marks:** 20
Exam Marks: 80	**Exam Duration:** 03

Course Outcomes (COs):

CO1 Explain software engineering concepts, process models, and agile practices.
CO2 Apply techniques to gather and manage software requirements.
CO3 Model systems using UML diagrams and system modelling techniques.
CO4 Design software architecture using patterns and UML notations.

Course Content	Hours
Unit 1	
OVERVIEW: Introduction; Software Engineering Ethics; Software Process Models; Process Activities; Coping with Change; Agile Software Development: Agile Methods; Plan-Driven and Agile Development.	11
Unit 2	
REQUIREMENTS ENGINEERING: Functional and Non-Functional Requirements; Software Requirements Document; Requirements Specification; Requirements Engineering Processes; Requirements Elicitation and Analysis; Requirements Validation; Requirements Management.	11
Unit 3	
SYSTEM MODELING: Context Models; Interaction Models- Use Case Modelling, Sequence Diagrams; Structural Models- Class Diagrams, Generalization, Aggregation, Behavioral Models-Data-Driven Modelling, Event-Driven Modelling; Model-Driven Engineering.	11
Unit 4	
ARCHITECTURAL DESIGN: Architectural Design Decisions; Architectural Views; Architectural Patterns- Layered Architecture, Repository Architecture, Client–Server Architecture Pipe and Filter Architecture. **DESIGN AND IMPLEMENTATION:** Object-Oriented Design Using The UML- System Context and Interactions, Architectural Design, Object Class Identification, Design Models, Interface Specification; Design Patterns; Implementation Issues.	11

Text Books:

1. Software Engineering by Ian Sommerville, Pearson publications;10th Edition, 2015.
2. A Concise Introduction to Software Engineering, Pankaj Jalote, 2nd Edition, Springer Cham, 2025.

Reference:

1. Software Engineering, N.S. Gill, Khanna Publishing House, 2023.
2. Software Engineering A Practitioner's Approach, 8th edition, Roger S Pressman, Bruce R. Maxim. McGraw Hill Education, 2015.

Semester: III

Course Code: GCS 304 **(Elective 2)**	**Course Title: E-commerce and E-Governance**
Course Credits: 03(3-0-0)	**Hours/Week:** 03
Total Contact Hours: 44	**Formative Assessment Marks:** 20
Exam Marks: 80	**Exam Duration:** 03

Course Outcomes (COs):

CO1: Understand the fundamentals, models, and infrastructure of E-commerce.

CO2: Analyze E-commerce strategies, marketing, and real-world applications.

CO3: Explain E-Governance models, technologies, and service frameworks.

CO4: Evaluate challenges, innovations, and future trends in E-Governance.

Course Contents	Hours
Unit-1	
Introduction to E-commerce: Fundamentals of E-commerce, E-commerce Business Models, E-commerce Infrastructure, E-commerce Security and Payment Systems	11
Unit-2	
E-commerce Strategies and Applications: E-commerce Marketing and Advertising, Supply Chain Management in E-commerce, Emerging Trends in E-commerce, Case Studies and Practical Applications.	11
Unit-3	
Introduction to E-Governance: Fundamentals of E-Governance, E-Governance Models and Frameworks, Technology and Infrastructure for E-Governance, E-Governance Services and Applications.	11
Unit-4	
Challenges and Future Trends in E-Governance: Challenges in E-Governance, E-Governance in Developing Countries, Emerging Technologies in E-Governance, Future Directions and Innovations.	11

Textbooks:

1. "E-Commerce 2020: Business, Technology, and Society" by Kenneth C. Laudon and Carol Guercio Traver, Pearson.
2. "Electronic Commerce 2018: A Managerial and Social Networks Perspective" by Efraim Turban, Jon Outland, David King, Jae Lee, Ting-Peng Liang, and Deborrah C. Turban, Springer.

Reference Books:

1. "E-Government: Information, Technology, and Transformation" by Hans J. Scholl, Routledge.
2. "E-Governance: Managing or Governing?" by Jeremy Millard, Routledge.
3. "Public Information Technology and E-Governance: Managing the Virtual State" by G. David Garson, Jones & Bartlett Learning.

Semester: IV

Course Code: GCS 401	**Course Title: Object Oriented Programming with Java**
Course Credits: 03 (3-0-0)	**Hours/Week:** 03
Total Contact Hours: 44	**Formative Assessment Marks:** 20
Exam Marks: 80	**Exam Duration:** 03

Course Outcomes (COs):
CO1: Understand OOP concepts and basic Java programming structure.
CO2: Apply classes, objects, constructors, arrays, and strings in Java.
CO3: Implement multithreading and exception handling in Java programs.
CO4: Perform file I/O and develop GUI applications using AWT and event handling.

Course Contents	Hours
Unit 1	
Fundamentals of Object-oriented Programming: Object-oriented Paradigm, Basic Principles of Object-oriented Programming, Advantages of Object-Oriented Programming, Applications of Object-Oriented Programming. **Introduction to Java Language:** Java History, Features, Overview, Difference between C, C++ and Java, Java Environment- JDK, JVM, JRE and API, Java Program Structure, Java Tokens, Implementing a Java Program, Command Line Arguments. **Java Programming Fundamentals:** Data types, Variables & Constants, Keywords & Naming Conventions, Type Casting, Operators and Expressions, Control Structures, Jumping Statements.	11
Unit 2	
Classes & Objects: Basics of Objects and Classes, Constructors, Access Modifiers, Method Overloading, Overloading Constructors, Static members, this keyword. **Arrays:** One-dimensional Arrays, Two-dimensional Arrays, Array of Objects. **Strings**: String Handling functions.	11
Unit-3	
Multithreading in Java: Concepts of Thread, Thread Life Cycle, Creating Threads & Implementing Runnable Interface, Thread Synchronization & Thread Priority. **Exception Handling:** Concepts of Exception, Different Types of Exceptions, Creating User-Defined Exceptions Using Try-Catch-Finally-Throw Blocks, Nested Try, Catch, Throw, and Throws Blocks.	11
Unit-4	
File Handling: I/O Handling, I/O Streams, Types of Files, Byte Stream, Binary I/O Classes & Its Hierarchy, FileInputStream & FileOutputStream Classes, Object I/O Classes. **Event Handling & GUI programming:** Event Handling, Event Types, Event Handling Mechanism, Keyboard & Mouse Handling, Introduction to AWT & GUI basics, AWT hierarchy of classes, AWT controls – Frames, Panels, Layout managers & other controls of AWT.	11

Text Books:
1. D.S. Guru, M.T. Somashekara, & K.S. Manjunatha, Object Oriented Programming with Java, PHI Learning, 2017.
2. E Balagurusamy, Programming with JAVA, TMH, 2007
   Reference Books: 1. Herbert Schildt, Java 7, The Complete Reference, , 8th Edition, 2009

Semester: IV

Course Code: GCS 402	**Course Title:** Java Lab
Course Credits: 02 (0-0-2)	**Hours/Week:** 04
Total Contact Hours: 60	**Formative Assessment Marks:** 10
Exam Marks: 40	**Exam Duration:** 03

Course Outcomes (COs):

CO1: Write Java programs using basic logic and structures.
CO2: Use OOP concepts like classes, objects, and inheritance.
CO3: Handle exceptions using Java's try-catch mechanisms.
CO4: Create GUI and file handling programs using AWT and I/O.

Laboratory Program List
Part A

1. Program to find whether the given number is Positive, Negative or Zero.
2. Program to list the factorial of the numbers 1 to 10.
3. Program to demonstrate classes & objects.
4. Program to demonstrate method overloading.
5. Program to demonstrate single inheritance (simple calculator – base class, Advanced Calculator – derived class).
6. Program to find Maximum & Minimum element in one dimensional array of numbers.
7. Program to check whether the given string is palindrome or not.
8. Program to create a 'Student' class with Reg.no., name and marks of 3 subjects. Calculate the total marks of 3 subjects and create an array of 3 student objects & display the results.

Part B

1. Program to generate negative array size exception
2. Program to generate NullPointer Exception.
3. Program that reads two integer numbers for the variables a and b. The program should catch NumberFormatException & display the error message.
4. Program to create AWT window with 4 buttons M/A/E/Close. Display M for Good Morning, A for Afternoon, E for evening and Close button to exit the window.
5. Program to demonstrate the various mouse handling events.
6. Program to read and write Binary I/O file.
7. Program to create window with three buttons father, mother and close. Display the respective details of father and mother as name, age and designation using AWT controls.
8. Program to create menu bar and pull-down menus.

Evaluation Scheme for Lab Examination

Assessment Criteria Marks		Marks
Writing	One Program from Part A	15
	One Program from Part B	15
Execution	Any one of the Written Program	5
Viva Voce based on Object Oriented with java		5
Total		**40**

Semester: III

Course Code: GCS 403 **(Elective 1)**	Course Title: Software Testing
CourseCredits:03(3-0-0)	**Hours/Week:**03
TotalContactHours:44	**Formative Assessment Marks:** 20
ExamMarks:80	**ExamDuration:**03

Course Outcomes (COs):

CO1: Understand the basics of software testing and test case design.

CO2: Apply decision table and data flow testing methods.

CO3: Analyze integration and system testing techniques.

CO4: Evaluate object-oriented and GUI testing approaches.

Course Content	Hours
Unit 1	
Basics of Software Testing and Examples: Basic definitions, Test cases, Insights from a Venn diagram, identifying test cases, Error and fault taxonomies, Levels of testing, Generalized pseudo code, The triangle problem, The Next Date function, The commission problem, The SATM (Simple Automatic Teller Machine) problem. **Decision Table-Based Testing:** Decision tables, Test cases for the triangle problem, Test cases for the Next Date function, Test cases for the commission problem.	11
Unit 2	
Data Flow Testing: Definition- Use testing, Slice-based testing, Guidelines and observations. **Life Cycle–Based Testing:** Traditional Waterfall Testing**,** Testing in Iterative Life Cycles**,** Agile Testing, **Model-Based Testing:** Testing Based on Models**,** Peterson's Lattice**,** Expressive Capabilities of Mainline Models**,** Modelling Issues**,** Making Appropriate Choices. **Integration Testing:** Introduction, Decomposition-based, call graph- based, Path-based integrations.	11
Unit 3	
System Testing: Definition, Possibilities, Basic concepts for requirements specification, Model-Based Threads, Use Case–Based Threads, ASF (Atomic System Functions). **Object-Oriented Testing:** Units for object-oriented testing, Implications of composition and encapsulation, inheritance, and polymorphism, Levels of object-oriented testing, GUI testing, Dataflow testing for object-oriented software. **Class Testing:** Methods as units, Classes as units.	11
Unit-4	
Object-Oriented Integration Testing: UML support for integration testing, MM paths for object-oriented software, A framework for object-oriented dataflow integration testing. **Object-Oriented System Testing:** Currency converter UML description, UML-based system testing, State chart-based system testing. **GUI Testing:** The currency conversion program, Unit testing, Integration Testing and System testing for the currency conversion program, case study of windshield wiper.	11

Text Books:

1.   Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rd Edition, Uerbach Publications.
2.   Aditya P Mathur: Foundations of Software Testing, Pearson.

Reference Books:

1.   Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, 1st edition, John Wiley & Sons.
2.   Srinivasan Desikan, Gopalaswamy Ramesh: Software testing Principles and Practices.

Semester: IV

Course Code: GCS 404 (Compulsory Paper)	Course Title: Digital Marketing
Course Credits: 02(2-0-0)	Hours/Week: 02
Total Contact Hours: 30	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 02:30

Course Outcomes (COs):

CO1 Describe the basics, evolution, and channels of digital marketing.
CO2 Apply social media and email marketing strategies effectively.
CO3 Create content and mobile marketing plans with analytics.

Course Content	Hours
Unit 1	
Introduction to Digital Marketing: Overview of digital marketing, Evolution of digital marketing, Importance and benefits of digital marketing, Digital marketing channels and platforms **Digital Marketing Strategy and Planning:** Developing a digital marketing strategy, Setting goals and objectives, Budgeting, and resource allocation. Campaign planning and execution, Monitoring and adjusting digital marketing campaigns.	10
Unit 2	
Social Media Marketing: Overview of social media marketing, social media platforms and their features, Creating and optimizing social media profiles, social media content strategy, social media advertising and analytics **Email Marketing:** Introduction to email marketing, building an email list, Creating effective email campaigns, Email automation and segmentation, Email marketing metrics and analytics.	10
Unit 3	
Content Marketing: Understanding content marketing, Content strategy and planning, Content creation and distribution, Content promotion and amplification, Content marketing metrics and analytics. **Mobile Marketing:** Mobile marketing overview, Mobile advertising strategies, Mobile app marketing, Location-based marketing, Mobile marketing analytics.	10

Text Books:

1. "Digital Marketing Strategy: An Integrated Approach to Online Marketing" by Simon Kingsnorth.
2. "Email Marketing Rules: How to Wear a White Hat, Shoot Straight, and Win Hearts" by Chad S. White.

Reference Books:
1. "Content Inc.: How Entrepreneurs Use Content to Build Massive Audiences and Create Radically Successful Businesses" by Joe Pulizzi.
2. "Mobile Marketing: How Mobile Technology is Revolutionizing Marketing, Communications and Advertising" by Daniel Rowles.
3. "Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity" by Avinash Kaushik.

Practical Evaluation [50 Marks]
- **Internal assessment: C1= 05 Marks, C2 =05 Marks [Total = 10 Marks]**
- **Semester End Practical Examination C3=40 marks**
- **Writing:** One program from both Part A and Part B (10 Marks each): **10 + 10 = 20 Marks**
- **Debug and Execution:** (05 Marks each) **05 + 05 = 10 Marks**
- **Record: 05 Marks**
- **Viva:** 05 **Marks**

Theory Evaluation [100 Marks]
CIE, SEE and QP Pattern for Theory Courses:
- Total Lecture hours per paper: 44
- No. of Units 4 (11 Hours Each)
- Internal Assessment **C1 = 10 Marks, C2 = 10 Marks**
- Semester End Theory Exam **C3 = 80 Marks**

Question Paper Pattern
Instructions: Answer Part-A and Part-B:

Part-A

Answer any 10 out of 12 Questions (3 Questions drawn from each unit).
Each question carries 2 Marks. (10 X 2 =20) Q. No. 1 to Q. No. 12.

Part-B

Answer the following Questions.

13.   a)

                OR

    b)

14.   a)

                OR

    b)

15.   a)

                OR

    b)

16.   a)

                OR

    b)

 Each main Question with internal choice and carries 15 Marks. (4 X 15 =60)
(Each main question may split into sub-questions like (10 + 5, 8 +7, 5+5+5, ...) and with a maximum of 3 sub-questions)

Question paper pattern for Theory Courses (2 Credits):

Maximum Marks – 40

Part A:

Answer any 5 questions: (5x2=10 Marks)

Question No. 1 to 6

Part B: Answer the following questions: (10x3=30 Marks)

7. a) 7. b)

   or

7. c) 7. d)

8. a) 8. b)

   or

8. c) 8. d)

9. a) 9. b)

   or

9. c) 9. d)