Bachelor of Computer Applications

Semester–III

Course Code	Course Title	L	T	P	Credits
BCA-201A	**Software Engineering** (Pre-requisite: None)	3	0	0	3
		CIE	**SEE**		**Total**
		40	60		100

Course Outcomes (COs): At the end of this course, students will be able to	
BCA-201A.1	Describe the phases of software development life cycle for designing an efficient software.
BCA-201A.2	Analyze requirement techniques like Data flow diagram, Entity relationship diagram, Object diagram
BCA-201A.3	Identification of user requirements using various requirements elicitation techniques
BCA-201A.4	Describe the basics of software design using various techniques.

Course Outcomes (CO) to Program Outcomes (PO) mapping (scale 1: Low, 2: Medium, 3: High)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
BCA-201A.1	3	-	-	-	-	-	-	-
BCA-201A.2	3	3	-	2	-	-	-	2
BCA-201A.3	3	-	-	2	-	-	-	2
BCA-201A.4	3	2	3	2	-	-	2	2

Instructions for Paper Setter: The examiner will set 9 questions asking two questions from each unit and one compulsory question. The compulsory question (Question No. 1) will consist at least 4 parts covering entire syllabus. The examinee will be required to attempt 5 questions, selecting one question from each unit and the compulsory question. All questions will carry equal marks.

UNIT-I **Contact Hours: 10**

Introduction to Software Engineering

Introduction: Software, Program, Software Crisis and Solutions, Software Evolution, Software Development Paradigm, Need of Software Engineering, Software Processes, Software Characteristics, Attributes of Software Product, Software Myths, Software basic terminologies, Characteristics of good software.

UNIT-II **Contact Hours: 13**

Software life cycle, Requirements Analysis and Specifications

Software life cycle models: Build and Fix, Waterfall, Prototype, Iterative Enhancement, Evolutionary Development Model, Prototyping, and Spiral Model.

Software Requirements Analysis and Specifications: Types of Requirements. Requirement Elicitation: Interview, Brainstorming, Quality Functional Deployment, Use Case Approach. Problem Analysis – DFD, Data dictionaries, ER diagrams, object diagrams, approaches to problems analysis, SRS, specifying behavioural & non-behavioural requirements.

UNIT-III **Contact Hours: 12**

Software Design

Software Design: Design framework, Conceptual and Technical Design, Trade-off between modularity and software cost, Cohesion and Coupling, Types of cohesion and coupling, Object oriented Design, Function Oriented Design.

UNIT-IV **Contact Hours: 10**

Software Metrics and Testing

Software Metrics: Definition, Categories of metrics, Token Count, Data Structure Metrics.

Software Testing: Introduction, difference between Bug, Mistake, error, Fault and Failure, Alpha, Beta and Acceptance Testing, Levels of Testing..

Text Books:

1. K. K. Aggarwal and Yogesh Singh, *Software Engineering*, New Age International Private Limited; Fourth edition.
2. R. S. Pressman, *Software Engineering – A practitioner's approach*, , McGraw Hill

Reference Books:

1. Richard Fairley, *Software Engineering Concepts*, McGraw Hill Education.
2. Pankaj Jalote, *An Integrated Approach to Software Engineering*, Narosa Publications

Course Code	Course Title	L	T	P	Credits
BCA-207A	Data Structures (Pre-requisite: C++ Programming)	3	0	0	3
		CIE	SEE		Total
		40	60		100

Course Outcomes (COs): At the end of this course, students will be able to

BCA-207A.1	Understand the Data Structures concepts
BCA-207A.2	Explore the operations of Arrays and Strings
BCA-207A.3	Examine the features and operations of Stack and Queue data structure.
BCA-207A.4	Explore the implementation and operations of linked lists applying searching and sorting algorithms.

Course Outcomes (CO) to Program2 Outcomes (PO) mapping (scale 1: Low, 2: Medium, 3: High)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
BCA-207A.1	3	-	-	-	-	-	-	-
BCA-207A.2	3	2	-	-	-	-	-	2
BCA-207A.3	3	2	-	2	-	-	-	2
BCA-207A.4	3	2	2	2	-	-	-	2

Instructions for Paper Setter: The examiner will set 9 questions asking two questions from each unit and one compulsory question. The compulsory question (Question No. 1) will consist at least 4 parts covering entire syllabus. The examinee will be required to attempt 5 questions, selecting one question from each unit and the compulsory question. All questions will carry equal marks.

UNIT-I **Contact Hours: 10**

Introduction to Data Structures

Basic data Structure: Introduction to Data Structures, algorithms and Pseudo code, Types of Data Structures, Relationship among data, data structures, and algorithms, Abstract Data Types, Analysis of Algorithms, asymptotic notations, asymptotic analysis.

UNIT-II **Contact Hours: 11**

Arrays and Strings

Arrays, Representation of Arrays in Memory, Operations on Array (Traversing, Insertion, Deletion), Sparse Matrices, Strings and its Representation in Memory, Operations on Strings (length, reverse, concatenation).

UNIT-III **Contact Hours: 12**

Stacks and Queues

Stacks: Introduction, Representation of Stacks Using Sequential Organization (Arrays), Applications of Stack, Expression Evaluation and Conversion, Polish notation and expression conversion, Processing of Function Calls, Reversing a String with a Stack, Recursion. Stack Abstract Data Type

Queues: Concept of Queues, Realization of Queues Using Arrays, Circular Queue, Multi-queues, Dequeue, Priority Queue, Applications of Queues, Queue as Abstract Data Type

UNIT-IV **Contact Hours: 12**

Linked Lists, Searching and Sorting techniques

Linked Lists: Introduction, Types (singly, double, circular, circular double) and Operations (Insertion, Deletion, Traversal, Searching, Sorting), Applications.

Searching and Sorting: Search Techniques-Linear Search, Binary Search, Sorting Techniques- Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort, Comparison of Sorting Methods

Text Books:

1. Michael T. Goodrich, R. Tamassia and Mount, *Data structures and Algorithms in C++*, John Wiley and Sons.
2. Mark Allen Weiss, *Data structures and Algorithm Analysis in C++*, Pearson Education.
3. Robert L. Kruse and A.J. Ryba, *Data Structures and Program Design in C++*, Prentice Hall, Inc., NJ

Reference Books:

1. Narasimha Karamanchi, *Data Structures and Algorithms Made Easy,* CareerMonk Publications
2. Adam Drozdek, *Data Structures and Algorithms in C++*, Course Technology.
3. Balaguruswami, E., *Object Oriented Programming In C++*, Tata McGraw-Hill.

Course Code	Course Title	L	T	P	Credits
BCA-182A	Data Structures Lab (Pre-requisite: C++)	0	0	4	2
		CIE	SEE		Total
		50	50		100

Course Outcomes (COs): At the end of this course, students will be able to

BCA-182A.1	Write programs in C++ implementing various operations on Array
BCA-182A.2	Implement various operations on Linked lists.
BCA-182A.3	Write programs implementing various operations on Stack and Queue
BCA-182A.4	Implement various Searching and Sorting techniques in C++.

Course Outcomes (CO) to Program Outcomes (PO) mapping (scale 1: Low, 2: Medium, 3: High)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
BCA-182A.1	3	2	-	2	-	-	-	2
BCA-182A.2	3	2	2	2	-	-	-	2
BCA-182A.3	3	2	2	2	-	-	-	2
BCA-182A.4	3	2	2	2	-	-	-	2

List of Experiments

No.	Experiment Detail
1.	Write a program to remove the duplicates from an array.
2.	Write a program that prints a table indicating the number of occurrences of each alphabet in the text entered.
3.	Write a program that uses functions to perform the following operations on an array (a) Creation (b) Insertion (c) Deletion (d) Traversal.
4.	Write a program to implement simple Stack using array to perform the following operations: (a) Create (b) Peek (c) Push (d) Pop (e) Traverse
5.	Write a menu driven program that implements singly linked list for the following operations: (b) Creation (b) Insertion (c) Deletion (d) Traversal.
6.	Write a menu driven program that implements doubly linked list for the following operations: Create, Display, Insert, Delete, Search,
7.	Write a program to implement simple Stack using Link list to perform the following operations: (a) Create (b) Peek (c) Push (d) Pop (e) Traverse
8.	Write a program to implement simple Queue using array to perform the following operations: (a) Create (b) Enqueue (c) Dequeue (d) Traverse
9.	Write a program to implement circular Queue using array to perform the following operations: (a) Create (b) Enqueue (c) Dequeue (d) Traverse
10.	Write a program to search an elements in an array using linear search technique.
11.	Write a program to search an elements in an array using binary search technique.
12.	Write a program to sort the elements using Insertion Sort algorithm.

13.	Write a program to sort the elements using Bubble Sort algorithm.
14.	Write a program to sort the elements using Quick Sort algorithm.
15.	Write a program to sort the elements using Merge Sort algorithm.
16.	Write a program to sort the elements using Selection Sort algorithm.

Text Books:

1. Michael T. Goodrich, R. Tamassia and Mount, *Data structures and Algorithms in C++*, John Wiley and Sons.
2. Mark Allen Weiss, *Data structures and Algorithm Analysis in C++*, Pearson Education.
3. Robert L. Kruse and A.J. Ryba, *Data Structures and Program Design in C++*, Prentice Hall, Inc., NJ
4. Herbert Scildt, *C++ The Complete Reference*, Tata McGraw-Hill

Reference Books:

1. Narasimha Karamanchi, *Data Structures and Algorithms Made Easy,* CareerMonk Publications
2. Adam Drozdek, *Data Structures and Algorithms in C++*, Course Technology.
3. Balaguruswami, E., *Object Oriented Programming In C++*, Tata McGraw-Hill.

Course Code	Course Title	L	T	P	Credits
BCA-203A	Database Management System (Pre-requisite: None)	3	0	0	3
		CIE	SEE		Total
		40	60		100

Course Outcomes (COs): At the end of this course, students will be able to	
BCA-203A.1	Understand the fundamentals of DBMS, data models, schema architecture, and ER/EER design.
BCA-203A.2	Apply SQL and PL/SQL concepts to create, query, and manage relational databases.
BCA-203A.3	Normalize database schemas using functional dependencies and various normal forms.
BCA-203A.4	Analyze transaction processing, concurrency control, and data warehousing concepts.

Course Outcomes (CO) to Program Outcomes (PO) mapping (scale 1: Low, 2: Medium, 3: High)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
BCA-203A.1	3	-	-	-	-	-	-	-
BCA-203A.2	3	2	2	2	-	-	-	-
BCA-203A.3	3	2	2	2	-	-	-	2
BCA-203A.4	3	3	-	2	-	2	2	3

Instructions for Paper Setter: The examiner will set 9 questions asking two questions from each unit and one compulsory question. The compulsory question (Question No. 1) will consist at least 4 parts covering entire syllabus. The examinee will be required to attempt 5 questions, selecting one question from each unit and the compulsory question. All questions will carry equal marks.

UNIT-I — Contact Hours: 11

Database and Database Management System

Introduction and need of Database and Database Management System (DBMS). Characteristics of DBMS, Database Users, Data Models, Schema, and Instances. Three-Schema architecture and data Independence.

Database Design using ER model: Entity types, Entity set, attributes and Keys. Relationship types. Expressing M:N relation. Enhanced Entity Relationship Model (EER): Specialization, Generalization, Attribute inheritance and Aggregation

UNIT-II — Contact Hours: 12

Relational model and Relational Algebra

Relational model concepts. Introduction to SQL, Types of SQL commands (DDL, DML, and DCL etc),SQL operators and their procedure, views and indexes. Queries and sub queries, Group by and Having clause, Aggregate functions and Constraints. PL/SQL: Architecture of PL/SQL, Basic Elements of PL/SQL.

Relational Algebra: Introduction of Relational Algebra, Selection and projection. Renaming, Joins, Unions, Intersection, Cartesian product and Division.

UNIT-III — Contact Hours: 10

Normalization

Need for Normalization, anomalies (insert, delete and update), Functional Dependencies, Minimal set of

Functional Dependencies, Normal Forms: 1NF, 2NF, 3NF. Higher Level Normal Forms: Boyce/ Codd Normal Form, multi-valued dependency (MVD), Fourth Normal Form, Join dependencies and Fifth Normal Form.

UNIT-IV **Contact Hours: 12**

Transaction processing, Overview of data mining and warehousing

Transaction Processing Concepts: Transaction and Schedules, transaction properties, concurrent execution of transaction, Conflict and View serializability, testing for serializability, concepts in Recoverable and Cascadeless schedules.Concurrency Control Techniques: Lock based protocols, Two phase locking technique, time stamp based protocols, validation based protocols.

Overview of data mining technology, Introduction of data warehousing, Characteristics of data warehouses.

Text Books:

1. Ramez Elmasri and Shamkant B. Navathe, *Fundamentals of Database Systems*, Pearson
2. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, *Database System Concepts*, McGraw Hill
3. Ivan Bayross, *SQL, PL/SQL the Programming Language of Oracle*, BPB

Reference Books:

1. Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw Hill
2. Markus Winand, *SQL Performance Explained*, Self Publishing

Course Code	Course Title	L	T	P	Credits
BCA-273A	Database Management System Lab (Pre-requisite: DBMS, PL/SQL)	0	0	4	2
		CIE	SEE		Total
		50	50		100

Course Outcomes (COs): At the end of this course, students will be able to

BCA-273A.1	Understand the fundamentals ER design.
BCA-273A.2	Apply SQL and PL/SQL concepts to create, query, and manage relational databases.
BCA-273A.3	Perform pattern matching and other operations
BCA-273A.4	Write PL/SQL programs using selection, loop, exception and Normalization.

Course Outcomes (CO) to Program Outcomes (PO) mapping (scale 1: Low, 2: Medium, 3: High)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
BCA-273A.1	3	-	-	-	-	-	-	-
BCA-273A.2	3	2	3	2	-	2	2	-
BCA-273A.3	3	-	-	-	-	-	-	-
BCA-273A.4	3	2	3	2	-	-	2	2

List of Experiments

No.	Experiment Detail
1.	Design an ER diagram for a college database with students, departments, and courses.
2.	Convert the ER diagram into relational schema and identify primary, foreign keys.
3.	Create tables in a relational DBMS using SQL based on the ER schema.
4.	Execute basic SQL commands: DDL, DML, DCL on a sample database
5.	Demonstrate the use of Data Constraints.
6.	Write SQL queries using WHERE, BETWEEN, LIKE, IN, and ORDER BY.
7.	Use aggregate functions with GROUP BY and HAVING clauses.
8.	Demonstrate pattern matching and range searching functions.
9.	Demonstrate the use of ORACLE operators • Arithmetic operators • Comparison operators • Logical operators
10.	Demonstrate the use of DUAL table
11.	Demonstrate the use of INDEX in DBMS
12.	Perform different types of joins (INNER, OUTER).
13.	Write PL/SQL programs using variables, IF-THEN, loops, and exceptions.
14.	Identify anomalies and perform step-by-step normalization (1NF to 3NF) on a sample unnormalized relation.

15.	Demonstrate the concept of locking using a small transaction simulation.

Text Books:

1. Ramez Elmasri and Shamkant B. Navathe, *Fundamentals of Database Systems*, Pearson
2. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, *Database System Concepts*, McGraw Hill
3. Ivan Bayross, *SQL, PL/SQL the Programming Language of Oracle,* BPB

Reference Books:

1. Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw Hill
2. Markus Winand, *SQL Performance Explained*, Self Publishing

Course Code	Course Title	L	T	P	Credits
BCA-205A	Programming with Java (Pre-requisite: None)	3	0	0	3
		CIE	SEE		Total
		40	60		100

Course Outcomes (COs): At the end of this course, students will be able to	
BCA-205A.1	Discuss the basic features of Java language.
BCA-205A.2	Understand various object-oriented programming (OOP) concepts in Java
BCA-205A.3	Understand Java concepts of packages and exceptions handling.
BCA-205A.4	Apply the knowledge of Java input output steams, and multithreading.

Course Outcomes (CO) to Program Outcomes (PO) mapping (scale 1: Low, 2: Medium, 3: High)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
BCA-205A.1	3	-	-	-	-	-	-	-
BCA-205A.2	3	2	2	2	-	-	-	2
BCA-205A.3	3	-	2	2	-	-	-	2
BCA-205A.4	3	2	2	2	-	-	-	2

Instructions for Paper Setter: The examiner will set 9 questions asking two questions from each unit and one compulsory question. The compulsory question (Question No. 1) will consist at least 4 parts covering entire syllabus. The examinee will be required to attempt 5 questions, selecting one question from each unit and the compulsory question. All questions will carry equal marks.

UNIT-I — Contact Hours: 12

Fundamentals of Java

Introduction to Java: Java Features, Java Virtual Machine and bytecode, Basics of Java programming: syntax, variables data types, operators and expressions, statements

Control Flow: Decision-making statements (if, else-if, switch). looping statements (for, while, do-while) and branching statements (break, continue, return),

Arrays: Declaring, initializing and manipulating arrays, array operations

UNIT-II — Contact Hours: 12

Object oriented programming concepts of Java

Classes and Objects: Declaring Classes and creating Objects, constructors, Garbage Collection, finalize() Method, Access modifiers (public, private, protected, default), static and final modifier, ‘this’ keyword, Method overloading, Wrapper Classes.

Inheritance: Extending classes, Method Overriding, ‘super’ keyword, Abstract classes, Multiple Inheritance, Interfaces and Extending Interfaces

UNIT-III **Contact Hours: 11**

Working with Packages and handling Exceptions

Packages: Java API Packages, importing packages, creating a new packages and using classes from package

Exception Handling: Types of Errors, Understanding Exceptions, Built-in Exceptions, checked and unchecked exceptions, try-catch block, multiple catch clauses, nested try block, finally block, throw and throws keywords, user created exceptions.

UNIT-IV **Contact Hours: 10**

Input output streams and multithreading

Java I/O Streams: Character and Byte streams, Reading console Input using java.util.Scanner and Writing console Output, Reading from and writing to files using FileInputStream, and FileOutputStream, FileReader and FileWriter. Object serialization and de-serialization.

Multithreading: Process versus Threads, Creating threads using Thread class and Runnable interface, thread lifecycle methods, Thread Priorities.

Text Books:

1. Patrick Naughton, Herbert, Schild, *The Complete reference Java 2*, Tata Mc-Graw Hill.
2. E. Balaguruswamy *Programming with JAVA- A Primer*, Tata Mc-Graw Hill publication

Reference Books:

1. Nell Dale, Chip Weems, *Programming and Problem Solving with Java,* Jones and Bartlett Publishers
2. Harvey Deitel, Paul Deitel, *Java: How to Program*, Pearson

Course Code	Course Title	L	T	P	Credits
BCA-275A	**Programming with Java Lab** (Pre-requisite: Java)	0	0	4	2
		CIE	**SEE**		**Total**
		50	50		100

Course Outcomes (COs): At the end of this course, students will be able to	
BCA-275A.1	Apply basic java features, control statements, and arrays
BCA-275A.2	Construct object-oriented programs in Java
BCA-275A.3	Implement exceptions handling.
BCA-275A.4	Design Java programs for file handling, and multithreading

Course Outcomes (CO) to Program Outcomes (PO) mapping (scale 1: Low, 2: Medium, 3: High)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
BCA-275A.1	3	2	-	-	-	-	-	-
BCA-275A.2	3	2	2	2	-	-	-	2
BCA-275A.3	3	2	2	2	-	-	-	2
BCA-275A.4	3	2	3	2	-	-	-	2

List of Experiments

No.	Experiment Detail
1.	Write a Java program that converts temperatures between Celsius and Fahrenheit based on user input.
2.	Implement a Java program to perform matrix multiplication using arrays.
3.	Write Java program to find the largest and smallest elements in an array.
4.	Implement a Java Program to sort an array of integers using Bubble sort.
5.	Write a program to implement method overloading.
6.	Develop a Java program to implement inheritance by creating a base class Animal and derived classes like Dog and Cat.
7.	Write a Java program to demonstrate method overriding by implementing a base class Shape, and derived classes Circle and Rectangle.
8.	Write a program to handle exceptions using try-catch.
9.	Write a program to demonstrate the use of ‘throws’ keyword.
10.	Write a program to handle user defined exceptions.
11.	Write a program to perform read and write operations on files using FileInputStream and FileOutputStream.
12.	Write a program to perform read and write operations on files using FileReader and FileWriter.
13.	Write a program to create and run multiple threads.
14.	Write a program to implement thread life-cycle methods.
15.	Write a program to implement thread priorities.

Text Books:

1. Patrick Naughton, Herbert, Schild, *The Complete reference Java 2*, Tata Mc-Graw Hill.
2. E. Balaguruswamy *Programming with JAVA- A Primer*, Tata Mc-Graw Hill publication

Reference Books:

1. Nell Dale, Chip Weems, *Programming and Problem Solving with Java*, Jones and Bartlett Publishers
2. Harvey Deitel, Paul Deitel, *Java: How to Program*, Pearson