Shiksha Mandal's
Bajaj College of Science, Wardha
(Formerly known as Jankidevi Bajaj College of Science, Wardha)

SYLLABUS FOR B.Sc. (Computer Science) PART II Semester III
(Approved in BoS Meeting held on 13.03.2025 to be implemented from the academic Session 2025-26)

Aim of the Course

To understand the basics of various data structures like stacks, queues, linked lists, etc., and their implementation using C++; and to understand the fundamental concepts related to hashing techniques, trees and graphs.

Learning Objectives	**Course Learning Outcomes**
Students will try to learn:	After successful completion of the course, student will be able to:
1. To understand the fundamentals of Data Structures, algorithmic complexities, and string processing operations.	1. Understand basic concepts, compute algorithmic complexities, and perform various string operations using C++ programs.
2. To know arrays, records, and pointers; their memory representations, and various sorting and searching techniques.	2. Write simple C++ programs demonstrating the usage of arrays, records, pointers, sorting, and searching.
3. To understand linked lists, their types, and operations on linked lists.	3. Understand and implement linked lists and their associated operations using C++ programs.
4. To get acquainted with stacks, notations, and recursion.	4. Implement stacks and recursion using C++.
5. To know concepts associated with queues, their representations, operations, and hashing techniques.	5. Create C++ programs demonstrating queues and their operations.
6. To get familiarised with the principles of trees and graphs, and their implementation using various methods.	6. Write programs on tree and graph traversals.

B.Sc. PART II - SEMESTER III

Course Name: DATA STRUCTURES (Major)

Course code: UCS230T

Credits: 4 **No. of Hrs.: 60**

Course Outline

Unit I: Introduction: Basic Terminology, Elementary Data Organization, Data Structures, Data Structure Operations, Algorithms: Complexity, Time-Space Trade-off.

String Processing: Introduction, Basic Terminology, Storing Strings, Character Data Type, String Operations.

Unit II: Arrays, Records, and Pointers: Linear Array, Representation of Linear Array in Memory, Traversing Linear Array, Inserting and Deleting, Bubble Sort, Linear Search, Binary Search.

Records: Record Structures, Representation of Records in Memory, Parallel Array. Pointers: Pointer Arrays,

Unit III: Linked List: Linked List, Representation of Single, Double, Header, Circular Single and Double Linked list, All possible operations on Single and Double Linked List using Dynamic representation, Polynomial Representation and its Manipulation.

Unit IV: Stacks: Stacks terminology, Representation of Stacks in Memory, Operation on Stacks, Polish Notations, Translation of infix to postfix & prefix expression, Infix to Postfix Conversion, Evaluation of Postfix Expression, Recursion, Problems on Recursion, Quick Sort and Tower of Hanoi Problem.

Unit V: Queue: Representation of Queues in Memory, Circular Queue. Dequeue and Priority Queue, Operations of above Structure using Array and Linked Representation.

Sorting and Searching: Selection Sort, Insertion Sort, Merge Sort, Efficiency of Sorting Methods, Big-O Notations. Hash Tables, Hashing Technique, Collision Resolution Technique.

Unit VI: Trees: Basic Terminologies, Representation of Binary Trees in Memory, Traversing of Binary tree, Binary Search Tree, Operation on Binary Search Tree, Heap Tree, Operation on Heap Tree, Heap Sort Method.

Graphs: Basic Terminologies, Definition and Representation of Graphs in Memory: Linked List and Matrix Representation. Traversing graphs: BFS, DFS Method.

REFERENCE BOOKS:

1. Classical Data Structures: D. Samantha. PHI, New Delhi.
2. DATA STRUCTURE: LIPSCHUTZ SCHAUM OUTLINE SERIES
3. Data Structures Using C++: Y. Kanetkar
4. Data Structures Using C++: Tanenbaum
5. Data Structure (Algorithms & Programs) - Dr. Vishal M. Lichade Sigma Solution Publication
6. Data Structures by Tremblay Sorenson

Web Resources:

Students are advised to make use of the resources available on the Internet. Some useful links related to computer science are given below.

1. www.tutorialspoint.com/data_structures_algorithms
2. www.onlinetutorialspoint.com/data-structures/what-is-data-structures
3. www.researchgate.net/publication/3786159_Tutorial

B.Sc. PART II - SEMESTER III

Course Name: DATA STRUCTURES Practical (Major)

Course Code: UCS230P

Credits: 2 **No. of Hrs.: 60**

1. Program to demonstrate various string operations.
2. Program to demonstrate linear search.
3. Program to demonstrate binary search.
4. Program to demonstrate bubble sort.
5. Program to insert a node at the beginning, at the end and in the middle of the given linked list.
6. Program to delete a node at the beginning, at the end and in the middle of the given linked list.
7. Program to create a linked list of customer names and their telephone numbers. (Using Menu Driven and include features of adding a new Customer and deleting an existing Customer.)
8. Program to reverse a linked list.
9. Program to search a value in the given linked list.
10. Program to insert a node at the beginning, at the end or in the middle of a given doubly linked list.
11. Program to delete a node from the beginning, at the end or in the middle of a given doubly linked list.
12. Program to create, insert and delete a node in Circular linked list.
13. Program to push and pop an element into / from a stack implemented using linked list.
14. Program to push and pop an element into / from a stack implemented using Array.
15. Program to evaluate postfix expression.
16. Program to sort an array using quick sort.
17. Program to solve Towers of Hanoi problems using recursion.
18. Program to perform insertion and deletion operations in linear queues.
19. Program to perform insertion and deletion operations on circular queues.
20. Program to sort an array using: Insertion sort.
21. Program to sort an array using: Selection sort.
22. Program to insert an element in a binary search tree.
23. Program to traverse inorder of a binary tree.
24. Program to traverse the preorder of a binary tree.
25. Program to traverse the postorder of a binary tree.

B.Sc. PART II - SEMESTER III
Vocational Skill Course (VSC)- II

Course Name: Web Designing
Course Code: UCS232P

Credits: 2 (4 Pr) **No. of Practical Hrs.: 60**

Course Description:

This course provides the foundational knowledge of computers to all students.

Learning Objectives:

This course is designed to satisfy the following objectives:

- To introduce the fundamentals of the Internet, and the principles of web design.
- To construct basic websites using HTML and Cascading Style Sheets.
- To build dynamic web pages with validation using JavaScript objects and by applying different event-handling mechanisms.

Course Learning Outcomes:

After successful completion of the course students will be able to:

- Describe the concepts of the World Wide Web and the requirements of effective web design.
- Develop web pages using the HTML and CSS features with different layouts as per need of applications.
- Use the JavaScript to develop the dynamic web pages.

Unit I: HTML: Introduction, Objective, HTML Command Tags, URLs, links, new web page creation, main body of the text, putting headers, adding a paragraph, formatting text in HTML and font mechanism, Color settings, superscripts and subscripts, and other manipulations on text and paragraphs, using directory and menu lists, creation of links, inserting graphics, using images, all manipulations on tables and its display.

Unit II: Detailed working with forms, allowing visitors to upload files, and active images, working with frames & framesets, Frames handling, scroll bars, and alternatives to frames. Introduction to browsers, working with e-mail, parts of e-mail text, working with messages.

DHTML: using DHTML in Internet Explorer, heading and horizontal line, hidden message, the message at the center of the page, moving boxes, changeable box.

Unit III: Introduction to CSS, creating style sheets, common tasks with CSS, Colors, the font -family, font metrics, length units, absolute units, relative units, the pixel unit, percentages as values, keywords as values, various properties such as the font -size property, font -size property etc, Assigning classes, tags, and attributes for applying classes, applying classes to an HTML tag, applying classes to other document parts, the layer tag, CSS Tags

List of Practical:

1. Write an HTML code to display your education details in a tabular format.
2. Write an HTML code to display your CV on a web page.
3. Write an HTML code to create a Home page having three links: About Us, Our Services, and Contact Us. Create separate web pages for the three links.
4. Write an HTML code to create a login form. On submitting the form, the user should get navigated to a profile page.
5. Write an HTML code to create a Registration Form. On submitting the form, the user should be asked to login with these new credentials.
6. Write an HTML code to create your Institute website, Department Website, and Tutorial website for a specific subject.
7. Write an HTML code to illustrate the usage of the following:
    a. Ordered List
    b. Unordered List
    c. Definition List
8. Write an HTML code to demonstrate the usage of inline CSS.
9. Write an HTML code to demonstrate the usage of internal CSS.
10. Write an HTML code to demonstrate the usage of external CSS.
11. Write a HTML code to prompt for users name and display it on the screen.
12. Design HTML form for keeping student record and validate it.

References:

1. Internet and web design by R Bangia, Second edition, Firewall media.
2. Multimedia and Wed technology by R Bangia.
3. Internet and web designing by ITELS. (Macmillan)
4. Web Enabled Commercial Application Development Using HTML, DHTML, JS, Perl by Ivan Bayross.
5. Deitel, Deitel & Nieto, Internet and Worldwide Web How to Program, Pearson Education, PHI.
6. Internmet Programming with VBScript and Java Script. Kathleen Kalata, (Thomson Publication)
7. Programming the World Wide Web By. Robert W. Sebesta. (Pearson)

Syllabus of Semester III Computer Science Minor

(BSc Physics with Computer Science as Minor or

BSc Electronics with Computer Science as Minor or

BSc Mathematics with Computer Science as Minor)

(Approved in BoS meeting held on 13.03.2025 w.e.f. Academic Session 2025-26)

Aim of the Course

To understand the basics of various data structures like stacks, queues, linked lists, etc., and their implementation using C++; and to understand the fundamental concepts related to hashing techniques, trees and graphs.

Learning Objectives	**Course Learning Outcomes**
Students will try to learn:	After successful completion of the course, student will be able to:
1. To understand the fundamentals of Data Structures, algorithmic complexities, and string processing operations.	1. Understand basic concepts, compute algorithmic complexities, and perform various string operations using C++ programs.
2. To know arrays, records, and pointers; their memory representations, and various sorting and searching techniques.	2. Write simple C++ programs demonstrating the usage of arrays, records, pointers, sorting, and searching.
3. To understand linked lists, their types, and operations on linked lists.	3. Understand and implement linked lists and their associated operations using C++ programs.
4. To get acquainted with stacks, notations, and recursion.	4. Implement stacks and recursion using C++.
5. To know concepts associated with queues, their representations, operations, and hashing techniques.	5. Create C++ programs demonstrating queues and their operations.
6. To get familiarised with the principles of trees and graphs, and their implementation using various methods.	6. Write programs on tree and graph traversals.

B.Sc. PART II - SEMESTER III

Course Name: Minor-III DATA STRUCTURES

Course code: UCS231T

Credits: 4 **No. of Hrs.: 60**

Course Outline

Unit I: Introduction: Basic Terminology, Elementary Data Organization, Data Structures, Data Structure Operations, Algorithms: Complexity, Time-Space Trade-off.

String Processing: Introduction, Basic Terminology, Storing Strings, Character Data Type, String Operations.

Unit II: Arrays, Records, and Pointers: Linear Array, Representation of Linear Array in Memory, Traversing Linear Array, Inserting and Deleting, Bubble Sort, Linear Search, Binary Search.

Records: Record Structures, Representation of Records in Memory, Parallel Array. Pointers: Pointer Arrays,

Unit III: Linked List: Linked List, Representation of Single, Double, Header, Circular Single and Double Linked list, All possible operations on Single and Double Linked List using Dynamic representation, Polynomial Representation and its Manipulation.

Unit IV: Stacks: Stacks terminology, Representation of Stacks in Memory, Operation on Stacks, Polish Notations, Translation of infix to postfix & prefix expression, Infix to Postfix Conversion, Evaluation of Postfix Expression, Recursion, Problems on Recursion, Quick Sort and Tower of Hanoi Problem.

Unit V: Queue: Representation of Queues in Memory, Circular Queue. Dequeue and Priority Queue, Operations of above Structure using Array and Linked Representation.

Sorting and Searching: Selection Sort, Insertion Sort, Merge Sort, Efficiency of Sorting Methods, Big-O Notations. Hash Tables, Hashing Technique, Collision Resolution Technique.

Unit VI: Trees: Basic Terminologies, Representation of Binary Trees in Memory, Traversing of Binary tree, Binary Search Tree, Operation on Binary Search Tree, Heap Tree, Operation on Heap Tree, Heap Sort Method.

Graphs: Basic Terminologies, Definition and Representation of Graphs in Memory: Linked List and Matrix Representation. Traversing graphs: BFS, DFS Method.

REFERENCE BOOKS:

1. Classical Data Structures: D. Samantha. PHI, New Delhi.
2. DATA STRUCTURE: LIPSCHUTZ SCHAUM OUTLINE SERIES
3. Data Structures Using C++: Y. Kanetkar
4. Data Structures Using C++: Tanenbaum
5. Data Structure (Algorithms & Programs) - Dr. Vishal M. Lichade Sigma Solution Publication
6. Data Structures by Tremblay Sorenson

Web Resources:

Students are advised to make use of the resources available on the Internet. Some useful links related to computer science are given below.

1. www.tutorialspoint.com/data_structures_algorithms
2. www.onlinetutorialspoint.com/data-structures/what-is-data-structures
3. www.researchgate.net/publication/3786159_Tutorial

B.Sc. PART II - SEMESTER III

Course Name: Minor-III DATA STRUCTURES Practical

Course code: UCS231P

Credits: 2 **No. of Hrs.: 60**

1. Program to demonstrate various string operations.
2. Program to demonstrate linear search.
3. Program to demonstrate binary search.
4. Program to demonstrate bubble sort.
5. Program to insert a node at the beginning, at the end and in the middle of the given linked list.
6. Program to delete a node at the beginning, at the end and in the middle of the given linked list.
7. Program to create a linked list of customer names and their telephone numbers. (Using Menu Driven and include features of adding a new Customer and deleting an existing Customer.)
8. Program to reverse a linked list.
9. Program to search a value in the given linked list.
10. Program to insert a node at the beginning, at the end or in the middle of a given doubly linked list.
11. Program to delete a node from the beginning, at the end or in the middle of a given doubly linked list.
12. Program to create, insert and delete a node in Circular linked list.
13. Program to push and pop an element into / from a stack implemented using linked list.
14. Program to push and pop an element into / from a stack implemented using Array.
15. Program to evaluate postfix expression.
16. Program to sort an array using quick sort.
17. Program to solve Towers of Hanoi problems using recursion.
18. Program to perform insertion and deletion operations in linear queues.
19. Program to perform insertion and deletion operations on circular queues.
20. Program to sort an array using: Insertion sort.
21. Program to sort an array using: Selection sort.
22. Program to insert an element in a binary search tree.
23. Program to traverse inorder of a binary tree.
24. Program to traverse the preorder of a binary tree.
25. Program to traverse the postorder of a binary tree.

B.Sc. PART II - SEMESTER III
Skill Enhancement Course (SEC) – II
Course Name: **JavaScript**
Course Code: UCS233P

Credits: 2 (4 Pr) **No. of Practical Hrs.: 60**

Course Description: This course introduces JavaScript web programming language. Learners would gain the fundamental knowledge required for understanding JavaScript.

Learning Objectives:

- To introduce fundamental concepts like data types, keywords, variables, conditionals, loops, and functions.
- To understand the principles of functions. Learn how to create and use functions effectively.
- To Learn how to use various objects (array, string, math, date) in JavaScript.

Course Learning Outcomes:

- Identify core aspects of programming and features of JavaScript.
- Understand and apply core concepts like data types, keywords, variables, conditionals, and loops.
- Use of various objects for writing and running JavaScript.

Unit- I: What is JavaScript? What is the evolution of JavaScript, what are its features, what are the advantages and disadvantages of JavaScript, How does JavaScript work? Structure of a JavaScript program, How to write JavaScript in Notepad?, How to add JavaScript in HTML?, JavaScript Comments, JavaScript Keywords, Data Types in JavaScript, JavaScript Variables, Types of Variables in JavaScript Key Differences between Var, Let, and Const.

Unit- II: Operators in JavaScript: Assignment, Comparison, Logical, Conditional, Bitwise, Unary, TypeOf, Operator Precedence. Conditional Statement: If statement, If else statement, Switch statement.

Loop statements: While Loop, Do While Loop, For loop, Nested for loops, For In loop, For Of loop, Break statement, Continue statement.

Unit- III: Functions in JavaScript, call a function in JavaScript using arguments, Pass By Value in JavaScript, return statement.

What is an Object? Types of Objects, Array Object, Properties of Array Object, Methods of Array Object.

String Object: Properties of String Object, Methods of String Object, Math Object, Properties of Math Object, Methods of Math Object, Date Object, Methods of Date Object.

List of Practical:

1. Write a JavaScript program to print Hello World.
2. Write a JavaScript Program to add two numbers.
3. Write a JavaScript Program to calculate the area of a triangle.
4. Write a JavaScript Program to swap two variables.
5. Write a JavaScript Program to check if a number is positive, negative, or zero.
6. Write a JavaScript Program to check if a number is odd or even.
7. Write a JavaScript Program to find the largest among three numbers.
8. Write a JavaScript Program to check the prime number using a function.
9. Write a JavaScript Program to find the factorial of a number using a function.
10. Write a JavaScript Program to display the multiplication table using a function.
11. Write a JavaScript Program to check if an array contains a specified value
12. Write a JavaScript Program to insert an item in an array.
13. Write a JavaScript Program to loop through an object.
14. Write a JavaScript Program to compare two strings.
15. Write a JavaScript Program to display the current Date.

References:

1. "JavaScript: The Definitive Guide" by David Flanagan.
2. Eloquent JavaScript: A Modern Introduction to Programming" by Marijn Haverbeke.
3. "JavaScript and JQuery: Interactive Front-End Web Development" by Jon Duckett.
4. You Don't Know JS" (Series) by Kyle Simpson.
5. The Principles of Object-Oriented JavaScript" by Nicholas C. Zakas.
6. JavaScript: The Good Parts" by Douglas Crockford.

* * *

Shiksha Mandal's
Bajaj College of Science, Wardha
(Formerly known as Jankidevi Bajaj College of Science, Wardha)

SYLLABUS FOR B.Sc. (Computer Science) PART II Semester IV
(Approved in BoS Meeting held on 13.03.2025 to be implemented from the academic Session 2025-26)

Aim of the Course

This course imparts the basics of Core Java programming concepts and implementation of AWT components as well as advanced topics like database and Swing.

Learning Objectives	Course Learning Outcomes
Students will try to learn:	After successful completion of the course, student will be able to:
1. To introduce Java programming and object-oriented concepts.	1. Learn the basic concepts of Java programming like variables, data types, operators, class, and object.
2. To understand object-oriented programming concepts, and apply them in solving problems.	2. Implement Java programs to solve problems.
3. To introduce the principles of arrays, strings, interfaces, and packages.	3. Understand and implement arrays, strings, interfaces, and packages in Java.
4. To get acquainted with concepts of applets, threads, and exceptions.	4. Write simple programs in Java on applets, threads, and exceptions.
5. To understand events, AWT components, and their controls in Java.	5. Design and implement GUI-based Java applications.
6. To know the techniques of layouts, JDBS, ODBC and Swing in Java.	6. Write programs to implement database connectivity and the use of Swing components.

B.Sc. PART II - SEMESTER IV

Course Name: DSC-IV JAVA PROGRAMMING (Major)

Course Code: UCS240T

Credits: 4 **No. of Hrs.: 60**

Course Outline

Unit I: Introduction to Java: History of Java, features of Java, getting started with Java. Java programs: Introduction of Application and Applets. **Variables:** Variable naming, variable initialization, assign values, Rules of variables, Scope of variable. **Operators:** Arithmetic, Assignment, Unary, Comparison, Shift, Bit- wise, Logical, Conditional, new, Special, Relational. **Data types:** Integers, Char, String, Float etc., Typecasting. **Tokens:** Java tokens, Order of precedence of operators, **Streams:** Input and output.

Unit II: Creating a class and subclass: Declaring a class, Naming a class, Rules to assign Class & Subclass, Creating a new object, Class of an object. **Data members:** Declaring data member, Naming variables, using class members. **Methods:** Using data members, Invoke a method, passing arguments to a method, calling method. **Access Specifiers and Modifiers:** Public, Private, Protected, Static and Final. **Overloading:** Method overloading, Constructor overloading. **Java class library:** Different types of classes.

Unit III: Decision-making and loops: If-then-else, Switch, ?: operator, While-loop, do-while loop, for loop. **Arrays:** Creating an array, one-dimensional array, two-dimensional array. **String:** String array, string methods. **Inheritance:** Single and multiple inheritances. **Interfaces:** Defining interfaces, extending interfaces, implementing interfaces.

Packages: Java API packages, creating packages, accessing packages, adding a class to packages. **Import statement:** Introduction, implementation of import statement.

Unit IV: Applets: Introduction to Applets & Application, how applets and applications are different, creating an applet. Applet life cycle, designing a web page, creating an executable applet, running the applet, applet tags, passing a parameter to an applet, HTML tag, Converting applet to application.

Threads: Overview of threads, single & multiple threads, life cycle of threads, stopping & blocking threads, working with threads, priority to thread, synchronization. **Exceptions & Errors:** Introduction, types of error, exception, syntax of exception, handling techniques, exception for Debugging.

Unit V: Events: Event driven programming, handling an (AWT) events. **Graphics class:** Introduction, the Graphics classes, drawing and filling of lines, rectangle, circle and ellipse, arcs, polygons, text and fonts, creating a font class, font objects, text, coloring object.

AWT Applications: Creating a GUI using AWT toolkit, using Component class, Frames. **Components and Control:** TextField, TextArea class, Label, Button, Choice, List, Checkbox, and Combo. **Menus:** Creating a popup menu.

Unit VI: Layouts: Using Window Listener interface, Different types of Layouts, Layout manager.

JDBC: Overview, JDBC implementation, Connection class, Statements, Catching Database Results, Handling Database Queries.

Swing: What is Swing? Difference between AWT and Swing, Swing Layouts, Managers, Containers, and Components – JFrame, JButton, JLabel, JText, JTextArea, JCheckBox And JRadioButton, JList, JComboBox, JMenu.

B.Sc. PART II - SEMESTER IV

Course Name: DSC-IV JAVA PROGRAMMING Practical (Major)

Course Code: UCS240P

Credits: 2 **No. of Hrs.: 60**

1. A) Write a program that declares a class, object and also it accesses the data member of it's class.
   B) Write an applet that accepts a value from the user and display it.

2. A) Write a program that accept marks of 5 subjects, calculates total, percentage, and display the grade according to their percentage.
   B) Write a program to print the multiplication table from 1 to 10.

3. A) Write a program to accept a set of values from the user into an array, display the values as well as their average.
   B) Accept string into a TextField, sort the characters in the string and display the sorted string into another TextField.

4. A) Write a program to demonstrate the overloading of constructor.
   B) Write an applet that accepts two numbers from user and display all the numbers between them.

5. A) Write a program to demonstrate the single inheritance.
   B) Write an applet to accept 10 numbers into an array, sort the array and display the sorted array. Accept the 10 numbers into the 10 different TextFields.
   C) Write a program to create a multiple selection list and also display the list of items selected by the user.

6. A) Write an applet to demonstrate the user menu bar.
   B) Write a sample program that will convert the applet to an application.
   C) Write a program to demonstrate the interfaces.

7. A) Write a program for exception handling that accepts two numbers as TextFields, the values are added and their sum is displayed. The code traps the error when user could enter text instead of numbers.
   B) Write a program that would accept input from the user and store it in a file called Test.java.

8. A) Write a program to implement the Graphics class. (Draw line, rectangle, fill rectangle, circle, oval).

B) Write an applet that displays a choice of the menu of three buttons (Add, Modify, Delete) selecting a choice from the menu should display the appropriate button. Use the show ( ) method of the Layout Manager.

9. Write a program to implement database connectivity.

10. A) Write a Java program to design and display the Swing controls.
    B) Write a program to implement the Swing Layout.

REFERENCE BOOKS:

1. The Art of Programming through Flowcharts & Algorithms by Anil B.Chaudhari, Firewall Media, Laxmi Publication, New Publication.
2. Programming with Java a Primer II edition: E Balagurusamy (TMH).
3. Java Programming (For Absolute Beginners) - Russell, PHI
4. Black Book on Java
5. Java- The Complete Reference by Herbert Schildt
6. Java (Theory & Programs) - Dr. Vishal M. Lichade Sigma Solution Publication

Web Resources:

Students are advised to make use of the resources available on the Internet. Some useful links related to computer science are given below.

1. www.tutorialspoint.com/cprogramming/
2. www.tutorialspoint.com
3. www.javatpoint.com/java-tutorial
4. www.tutorialspointexamples.com
5. www.w3schools.com

Syllabus of Semester IV Computer Science Minor

(BSc Physics with Computer Science as Minor or

BSc Electronics with Computer Science as Minor or

BSc Mathematics with Computer Science as Minor)

(Approved in BoS meeting held on 13.03.2025 w.e.f. Academic Session 2025-26)

Aim of the Course

This course imparts the basics of Core Java programming concepts and implementation of AWT components as well as advanced topics like database and Swing.

Learning Objectives	**Course Learning Outcomes**
Students will try to learn:	After successful completion of the course, student will be able to:
1. To introduce Java programming and object-oriented concepts.	1. Learn the basic concepts of Java programming like variables, data types, operators, class, and object.
2. To understand object-oriented programming concepts, and apply them in solving problems.	2. Implement Java programs to solve problems.
3. To introduce the principles of arrays, strings, interfaces, and packages.	3. Understand and implement arrays, strings, interfaces, and packages in Java.
4. To get acquainted with concepts of applets, threads, and exceptions.	4. Write simple programs in Java on applets, threads, and exceptions.
5. To understand events, AWT components, and their controls in Java.	5. Design and implement GUI-based Java applications.
6. To know the techniques of layouts, JDBS, ODBC and Swing in Java.	6. Write programs to implement database connectivity and the use of Swing components.

B.Sc. PART II - SEMESTER IV

Course Name: Minor-IV JAVA PROGRAMMING

Course code: UCS241T

Credits: 4 **No. of Hrs.: 60**

Course Outline

Unit I: Introduction to Java: History of Java, features of Java, getting started with Java. Java programs: Introduction of Application and Applets. **Variables:** Variable naming, variable initialization, assign values, Rules of variables, Scope of variable. **Operators:** Arithmetic, Assignment, Unary, Comparison, Shift, Bit- wise, Logical, Conditional, new, Special, Relational. **Data types:** Integers, Char, String, Float etc., Typecasting. **Tokens:** Java tokens, Order of precedence of operators, **Streams:** Input and output.

Unit II: Creating a class and subclass: Declaring a class, Naming a class, Rules to assign Class & Subclass, Creating a new object, Class of an object. **Data members:** Declaring data member, Naming variables, using class members. **Methods:** Using data members, Invoke a method, passing arguments to a method, calling method. **Access Specifiers and Modifiers:** Public, Private, Protected, Static and Final. **Overloading:** Method overloading, Constructor overloading. **Java class library:** Different types of classes.

Unit III: Decision-making and loops: If-then-else, Switch, ?: operator, While-loop, do-while loop, for loop. **Arrays:** Creating an array, one-dimensional array, two-dimensional array. **String:** String array, string methods. **Inheritance:** Single and multiple inheritances. **Interfaces:** Defining interfaces, extending interfaces, implementing interfaces.

Packages: Java API packages, creating packages, accessing packages, adding a class to packages. **Import statement:** Introduction, implementation of import statement.

Unit IV: Applets: Introduction to Applets & Application, how applets and applications are different, creating an applet. Applet life cycle, designing a web page, creating an executable applet, running the applet, applet tags, passing a parameter to an applet, HTML tag, Converting applet to application.

Threads: Overview of threads, single & multiple threads, life cycle of threads, stopping & blocking threads, working with threads, priority to thread, synchronization. **Exceptions & Errors:** Introduction, types of error, exception, syntax of exception, handling techniques, exception for Debugging.

Unit V: Events: Event driven programming, handling an (AWT) events. **Graphics class:** Introduction, the Graphics classes, drawing and filling of lines, rectangle, circle and ellipse, arcs, polygons, text and fonts, creating a font class, font objects, text, coloring object.

AWT Applications: Creating a GUI using AWT toolkit, using Component class, Frames. **Components and Control:** TextField, TextArea class, Label, Button, Choice, List, Checkbox, and Combo. **Menus:** Creating a popup menu.

Unit VI: Layouts: Using Window Listener interface, Different types of Layouts, Layout manager.

JDBC: Overview, JDBC implementation, Connection class, Statements, Catching Database Results, Handling Database Queries.

Swing: What is Swing? Difference between AWT and Swing, Swing Layouts, Managers, Containers, and Components – JFrame, JButton, JLabel, JText, JTextArea, JCheckBox And JRadioButton, JList, JComboBox, JMenu.

B.Sc. PART II - SEMESTER IV

Course Name: Minor-IV JAVA PROGRAMMING Practical

Course code: UCS241P

Credits: 2 **No. of Hrs.: 60**

1. A) Write a program that declares a class, object and also it accesses the data member of it's class.
   B) Write an applet that accepts a value from the user and display it.

2. A) Write a program that accept marks of 5 subjects, calculates total, percentage, and display the grade according to their percentage.
   B) Write a program to print the multiplication table from 1 to 10.

3. A) Write a program to accept a set of values from the user into an array, display the values as well as their average.
   B) Accept string into a TextField, sort the characters in the string and display the sorted string into another TextField.

4. A) Write a program to demonstrate the overloading of constructor.
   B) Write an applet that accepts two numbers from user and display all the numbers between them.

5. A) Write a program to demonstrate the single inheritance.
   B) Write an applet to accept 10 numbers into an array, sort the array and display the sorted array. Accept the 10 numbers into the 10 different TextFields.
   C) Write a program to create a multiple selection list and also display the list of items selected by the user.

6. A) Write an applet to demonstrate the user menu bar.
   B) Write a sample program that will convert the applet to an application.
   C) Write a program to demonstrate the interfaces.

7. A) Write a program for exception handling that accepts two numbers as TextFields, the values are added and their sum is displayed. The code traps the error when user could enter text instead of numbers.
   B) Write a program that would accept input from the user and store it in a file called Test.java.

8. A) Write a program to implement the Graphics class. (Draw line, rectangle, fill rectangle, circle, oval).

   B) Write an applet that displays a choice of the menu of three buttons (Add, Modify, Delete) selecting a choice from the menu should display the appropriate button. Use the show ( ) method of the Layout Manager.

9. Write a program to implement database connectivity.

10. A) Write a Java program to design and display the Swing controls.
    B) Write a program to implement the Swing Layout.

REFERENCE BOOKS:

1. The Art of Programming through Flowcharts & Algorithms by Anil B.Chaudhari, Firewall Media, Laxmi Publication, New Publication.
2. Programming with Java a Primer II edition: E Balagurusamy (TMH).
3. Java Programming (For Absolute Beginners) - Russell, PHI
4. Black Book on Java
5. Java- The Complete Reference by Herbert Schildt
6. Java (Theory & Programs) - Dr. Vishal M. Lichade Sigma Solution Publication

Web Resources:

Students are advised to make use of the resources available on the Internet. Some useful links related to computer science are given below.

1. www.tutorialspoint.com/cprogramming/
2. www.tutorialspoint.com
3. www.javatpoint.com/java-tutorial
4. www.tutorialspointexamples.com
5. www.w3schools.com

B.Sc. PART II - SEMESTER IV
Skill Enhancement Course (SEC) – III
Course Name: **MySQL**
Course Code: UCS243P

Credits: 2 (4 Pr) **No. of Practical Hrs.: 60**

Course Description: This course introduces the MySQL database. Learners would gain the fundamental knowledge required for understanding MySQL.

Learning Objectives:

- Understand basic statements of MySQL.
- know how to use MySQL database tool to connect to the database.
- Write MySQL statements to interact with the database.
- know how to join tables to address business problems.

Course Learning Outcomes:

- Identify core aspects of MySQL and its features.
- Understand and apply various statements of MySQL database.
- Use of various types of joins in MySQL.

Unit- I: Introduction of MySQL, Features, Advantages of MySQL, Installation, Data Types, Variables, Create User, Drop User, show Users, Change User Password, Create Database, Select Database, Show Database, Drop Database, Create Table, Drop Table, Alter Table, Show Table, Truncate Table, Temporary Table, Copy Table, Views

Unit- II: MySQL Queries, Constraints, insert Query, Update Query, Delete Query, Select Query, Insert On Duplicate Key, Update MySQL, Create Index, Drop Index, show Index, Clustered Index, Clustered vs Non-Clustered Index, MySQL Where AND OR LIKE Limit, Order By, Group By, Having, Distinct, Union, Union All, MySQL IF(), IFNULL() NULLIF(), CASE IF Statement.

Unit- III: MySQL Join, Inner Join, Outer Join, Cross Join, Self Join, Delete Join Union vs Join Primary Key, Foreign Key, Unique Key, Composite Key, MySQL count(), sum(), avg() max(), min(), first(), last(), GROUP_CONCAT(), MySQL Comments, Commands Cheat-Sheet Transaction, Cursor, Limit, Stored Function, Format Function, Ranking Function, Window Function, Union, Varchar, Enum, Set, Decimal, Cast, Convert() wildcards, Alias Function, Trigger.

List of Practical:

1. Introduction to MySQL, Database creation, and Table creation.
2. Data insertion, update/modification/Delete, and retrieval through MySQL. Basic SQL structure. Query implementation.
3. Enforcing integrity constraints (Domain, Key constraints (Primary/Foreign keys), NOT NULL, UNIQUE, DEFAULT, Check).
4. Creating and updating View. Query implementation using View.
5. Use of aggregate functions (AVG, COUNT, MIN, MAX, SUM).
6. Use of Join operator (Natural join, Outer join (left, right and full).
7. Query optimization through Nested Query (Use of logical connectives, set comparison operators, Union, Intersect, Except, Exists clauses).
8. Use of Group By and Having clause, Trigger creation.
9. Index creation through SQL.

References:

1. MySQL by Paul DuBois, 2000.
2. MySQL Cookbook: Solutions for Database Developers and Administrators by Paul DuBois, 2014.
3. MySQL and JSON: A Practical Programming Guide by David R. Stokes, 2018.
4. MySQL: The Complete Reference by Vikram Vaswani, 2003.
5. MySQL Explained: Your Step-by-Step Guide to Database Design by Andrew Comeau, 2015.
6. Learn SQL with MySQL: Retrieve and Manipulate Data Using SQL Commands with Ease (English Edition) by Pajankar Ashwin, 2020.
7. Murach's MySQL by Joel Murach, 2012.
8. Effective MySQL Replication Techniques in Depth by Ronald Bradford, 2012.

* * *