

Third Year B.C.A. (Under Science) Semester V

Course Code: BCA501

Course Title: Java Programming

**Total Contact Hours: 48 hrs.
(60 Lectures)**

Total Credits: 04

Total Marks: 100

Teaching Scheme: Theory- 05 Lect./ Week

Course Objectives:

The syllabus aims in equipping students with

- To understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
- To handle abnormal termination of a program using exception handling

To use the Java SDK environment to create, debug and run simple Java program

| | |
|--|-----------|
| Unit 1: Introduction to Java | 04 |
| <p>Basics of Programming Language</p> <ul style="list-style-type: none">• History and Features of Java• JDK,JRE,JIT and JVM• Naming Convention• Simple java program• Java IDE –Eclipse/ NetBeans (Note: For Lab Demonstration) <p>Introduction to Java</p> <p>Data Types Variable: final, static, abstract</p> <ul style="list-style-type: none">• Types of Comments• Array: 1D, 2D, Dynamic array using Vector• Accepting input using Command line argument• Accepting input from console (Using BufferedReader and Scanner class) | |
| Unit 2: Usage of Objects and Classes | 04 |
| <ul style="list-style-type: none">• Defining Your Own Classes• Access Specifiers (public, protected, private, default/friendly)• Array of Objects• Constructors, Overloading Constructors and use of ‘this’ Keyword• Predefined classes<ul style="list-style-type: none">➤ String class (Basic Functions) | |

| | |
|---|--|
| <ul style="list-style-type: none"> ➤ StringBuffer class ➤ Wrapper class • Inner classes, Nested classes, local classes, Anonymous classes(Anonymous object) • Introduction to Packages : Creation, Access and use • Garbage Collection (finalize() Method) | |
|---|--|

| | |
|---|-----------|
| Unit 3: Inheritance and Interface | 03 |
| <ul style="list-style-type: none"> • Inheritance Basics (extends Keyword) • Types of Inheritance • use of 'super' Keyword • Usage of final keyword related to method and class • Usage of abstract class and abstract methods • Interface: Defining and Implementing Interfaces • Runtime polymorphism using interface | |

| | |
|---|-----------|
| Unit 4: Collection | 07 |
| <ul style="list-style-type: none"> • Collection interface <ul style="list-style-type: none"> ➤ Collection framework ➤ Collection interfaces & classes-ArrayList, LinkedList, HashSet, TreeSet • Iterator and Enumeration, Hash Table. • Vector. | |

| | |
|---|-----------|
| Unit 5: Exception Handling and I/O | 10 |
|---|-----------|

| | |
|---|--|
| <ul style="list-style-type: none"> • Exception handling fundamentals • Exception types • Exception class <ul style="list-style-type: none"> ➤ Checked exception ➤ Unchecked exception • Creating user defined exception • Uncaught exceptions • Assertions • Introduction to Java.io package • Byte streams • Character streams • File IO basics • Object serialization – Reader and Writer | |
|---|--|

| | |
|---|-----------|
| Unit 6: Swing , Applet programming | 09 |
| <ul style="list-style-type: none"> • MVC(Model View Controller) Architecture • Swing components : JFrame, JPanel, JButton, JcheckBox, JTextField, JRadioButton, JLabel, JList, JDialog, JFileChooser, JColorChooser, JMenu • Applet fundamentals, Applet lifecycle, Creating and running applets • Applets: Event Handling using applets. | |

| | |
|--|----------|
| Unit 7: Database Programming | 8 |
| <ul style="list-style-type: none"> • Introduction to JDBC: Architecture (2-tier, 3-tier) • JDBC Drivers • Connectivity with PostgreSQL: basic steps • JDBC statement: Statement, PreparedStatement, CallableStatement • JDBC ResultSet and types • JDBC Metadata – ResultSetMetaData, DatabaseMetaData | |

| | |
|---|-----------|
| Unit 8: Servlets | 08 |
| <ul style="list-style-type: none"> • Introduction to Servlet and Servlet types | |

| | |
|---|--|
| <ul style="list-style-type: none"> • Lifecycle of servlet • Handling HTTPRequest and HTTPResponse • HttpServlet: <ul style="list-style-type: none"> ➤ Reading form data from servlet ➤ Servlet - Database communication • Session tracking –User Authorization, URL Rewriting, Hidden Form fields, Cookies and HttpSession | |
|---|--|

| | |
|---|----------|
| Unit 9: Java Server Pages (JSP) | 7 |
| <ul style="list-style-type: none"> • Introduction to JSP • Life cycle of JSP • Implicit Objects • Scripting elements –Declarations, Expressions, Scriptlets, Comments • JSP Directives – Page Directive, include directive • Basic JSP program • Mixing Scriptlets and HTML • Example of forwarding contents from database to servlet, servlet to JSP and displaying it using JSP scriptlet tag | |

| |
|--|
| Reference Books: |
| <ol style="list-style-type: none"> 1. Complete reference Java by Herbert Schildt(5th edition) 2. Java 2 Programming Black Book, Steven Horlzner 3. Programming with java, a Primer, 4th edition, By e balgurusamy 4. Core Java Volume I- Fundamentals, 8th edition, Cay S Horstmann,Gary Cornell, Prentice Hall, Sun MicroSystem Press 5. Core Java Volume II- Advance Features, 8th edition, Cay S Horstmann, Gary Cornell, Prentice Hall, Sun MicroSystem Press |

Third Year B.C.A. (Under Science) Semester V

Course Code: BCA 502

Course Title: Advanced Web Technology

Total Contact Hours: 48hrs.(60 lectures)

Total Credits: 04

Total Marks: 100

Teaching Scheme: Theory- 05 Lect. /Week

Objectives :-

1. To know & understand concepts of internet programming.

| Unit No | Contents | No Of Lectures |
|----------------|--|-----------------------|
| 1 | Introduction to Object Oriented Programming in PHP 1.1 Classes 1.2 Objects 1.3 Encapsulation 1.4 Constructor and Destructor 1.5 Inheritance 1.6 Interfaces 1.7 Introspection | 10 |
| 2 | Web Techniques 2.1 Super global Variables 2.2 Server information 2.3 Sticky forms 2.4 File Uploads 2.5 Setting response headers 2.6 Maintaining state 2.6.1. Session and Cookies | 12 |
| 3 | Files and Directories 3.1 Working with files and directories 3.2 Opening and Closing 3.3 Getting information about file 3.4 Reading and writing characters in file 3.5 Rename and delete files 3.6 Random access to file data 3.7 Getting information on file 3.8 Ownership and permissions | 10 |
| 4 | Databases(Postgresql) 4.1 Using PHP to access/insert/update/delete a database tables 4.2 Relational databases and SQL 4.3 Introduction to PEAR DB basics (No assignments) 4.4 Advanced database techniques 4.5 Simple applications | 9 |
| | XML 5.1 What is XML? | |

| | | |
|---|---|---|
| 5 | 5.2 XML document Structure 5.3 PHP and XML 5.4 XML parser 5.5 The document object model 5.6 The simple XML extension 5.7 Changing a value with simple XML | 8 |
| 6 | Ajax 6.1 Understanding java scripts for AJAX 6.2 AJAX web application model 6.3 AJAX –PHP framework 6.4 Performing AJAX validation 6.5 Handling XML data using php and AJAX 6.6 Connecting database using php and AJAX | 8 |
| 7 | Introduction to Web Services 7.1. SOAP 7.2. WSDL 7.3. Application of web services | 3 |

Reference Books : -

1. Complete HTML- Thomas Powell
2. HTML and JavaScript–Ivan Bayross
3. Programming PHP By Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication
4. Beginning PHP5, Wrox publication
5. PHP for Beginners, SPD publication

Third Year B.C.A. (Under Science) Semester V

Course Code: BCA 503

Course Title: Software Quality Assurance

**Total Contact Hours: 48 hrs.
(60 Lectures)**

Total Credits: 04

Total Marks: 100

Teaching Scheme: Theory- 05 Lect./ Week

Pre-requisites(if any) :

1. Basic concepts of Software Engineering

Course Objectives:

1. To understand the basic of quality software and quality factors.
2. To understand software quality architecture and component.
3. To understand software project life cycle, infrastructure and software quality standards.

| Unit No. | Contents | No. of Lectures |
|-----------------|--|------------------------|
| Unit 1 | 1. Introduction to Software Quality. 1.1. Uniqueness of software quality assurance 1.2. Software, Software errors, Faults and Failures 1.3. Classification of the causes of software errors 1.4. Software quality, Software quality assurance and software engineering | 06 |
| Unit 2 | 2. Software Quality Architecture and Components 2.1. The need for comprehensive software quality requirements 2.2. Classifications of software requirements into software quality factors 2.2.1. Product Operation 2.2.2. Product Revision 2.2.3. Product Transition 2.3. Parties interested in the definition of quality requirements. 2.4. SQA architecture 2.5. Software Quality Components 2.5.1. Pre-project components 2.5.2. Software project life cycle components 2.5.3. Infrastructure components for error prevention and improvement 2.6. Management SQA components | 10 |
| Unit 3 | 3. Project Life Cycle. 3.1. Classic and other software development methodologies 3.2. Factors affecting intensity of quality assurance activities in the development process 3.3. Verification, validation and qualification 3.4. A model for SQA defect removal effectiveness and cost | 14 |

| | | |
|--------|--|----|
| | <p>3.5. Demonstration of CASE study and CASE tools</p> <p>3.5.1. What is a CASE tool?</p> <p>3.5.2. The contribution of CASE tools to software product quality</p> <p>3.5.3. The contribution of CASE tools to software maintenance quality</p> <p>3.5.4. The contribution of CASE tools to improved project management</p> | |
| Unit 4 | <p>4. Software Quality Infrastructure Components</p> <p>4.1. Procedures and work instructions –</p> <p>4.1.1. Need</p> <p>4.1.2. Procedures manuals</p> <p>4.1.3. work instruction manuals</p> <p>4.1.4. Procedures and work instructions: preparation, implementation and updating</p> <p>4.2. Supporting Quality devices</p> <p>4.2.1. Templates</p> <p>4.2.2. Checklists</p> <p>4.3. Configuration management -</p> <p>4.3.1. Software configuration, its items and its management</p> <p>4.3.2. Software configuration management – tasks and organization</p> <p>4.3.3. Software change control</p> <p>4.3.4. Release of software configuration versions</p> <p>4.3.5. Provision of SCM information services</p> <p>4.3.6. Software configuration management audits</p> <p>4.3.7. Computerized tools for managing software configuration</p> | 10 |
| Unit 5 | <p>5. Software quality metrics</p> <p>5.1. Objectives of quality measurement</p> <p>5.2. Classification of software quality metrics</p> <p>5.3. Process metrics</p> <p>5.4. Product metrics</p> <p>5.5. Implementation of software quality metrics</p> <p>5.6. Limitations of software metrics</p> | 10 |
| Unit 6 | <p>6. Software Quality Standards, certification and assessment</p> <p>6.1. Quality management standards</p> <p>6.1.1. The scope of quality management standards</p> <p>6.1.2. ISO 9001 and ISO 9000-3</p> <p>6.1.3. Certification according to ISO 9000-3</p> <p>6.1.4. Capability Maturity Models – CMM and CMMI assessment methodology</p> <p>6.1.5. The Bootstrap methodology</p> <p>6.1.6. The SPICE project and the ISO/IEC 15504 software process assessment standard</p> <p>6.2. Project process standards</p> <p>6.2.1. Structure and content of IEEE software engineering standards</p> | 10 |

| | | |
|--|---|--|
| | 6.2.2. IEEE/EIA Std 12207 – software life cycle processes | |
| | 6.2.3. IEEE Std 1012 – verification and validation | |
| | 6.2.4. IEEE Std 1028 – reviews | |

Reference Books:

1. Software Quality Assurance by Daniel Galin, Pearson Publication, 2009.
2. Software testing and Quality Assurance Theory and Practice by Kshirasagar Naik and Priyadarshi Tripathy, Wiley Publication.
3. Software Engineering A Practitioner's Approach Sixth Edition by Roger S. Pressman, McGraw Hill Publication
4. Metrics and Models in Software Quality Engineering, By Stephen H. Kan, Pearson Publication

Third Year B.C.A. (Under Science) Semester V

Course Code: BCA504

Course Title: Operating Systems

**Total Contact Hours: 48 hrs.
(60 Lectures)**

Total Credits: 04

Total Marks: 100

Teaching Scheme: Theory- 05 Lect./ Week

Pre-requisites : Knowledge of fundamentals of Computer Organization

Course Objectives:

1. To understand the objectives, structure and functions of operating system
2. To learn about concept of processes, threads and its scheduling algorithms
3. To understand design issues in process synchronization and deadlock management
4. To study various memory management schemes
5. To learn about concept file and I/O management in detail.

| Unit No. | Content | No. of Lectures |
|---------------|---|-----------------|
| Unit 1 | 1. Introduction to Operating System Concepts 1.1 Operating System Objectives and Functions - Definition of Operating System ,Role and Objectives of Operating System, Operating System as a User View and System View 1.2 Evolution Of Operating Systems - Batch Operating System, Multi-Programming Operating System ,Time-Sharing Operating System, Desktop Operating Systems, Real-Time Operating System, Distributed Operating System, Parallel Systems, Multimedia Systems, Handheld Systems 1.3 Computer System Architecture - Single-Processor Systems, Multi-Processor Systems, Clustered Systems 1.4 Operating System Operations- Dual-Mode And Multimode Operation, Timer 1.5 Operating System as Resource Management- Process Management, Memory Management, Storage Management(File system ,Mass storage ,Caching I/O systems), Protection And Security 1.6 Computing Environment-Traditional Computing, Client Server Computing, Peer To Peer Computing, Virtualization, Cloud Computing | 06 |
| Unit 2 | 2. System structure 2.1 Operating System Services 2.2 System Calls Concepts 2.3 Types of system Calls- Process Control, File Management, Device Management, Information Maintenance, Communication, Protection 2.4 System Programs | 03 |

| | | |
|---------------|--|-----------|
| | 2.5 System Boot | |
| Unit 3 | 3. Process and Thread Management 3.1 Process Concept – Process , Process Model , Process Control Block 3.2 Operations on Process – Process creation , Process Termination 3.3 Process Scheduling - Scheduling queues, Schedulers, Context switch 3.4 Inter Process Communication –Cooperating Process, Shared Memory Systems, Message Passing Systems 3.5 Overview of Threads 3.6 Concept of Multithreaded Programming and Multicore Programming 3.7 Types of threads – User and Kernel 3.8 Multithreading Models – Many to One , One to Many, Many to Many | 06 |
| Unit 4 | 4 Process Scheduling 4.1 Basic Concept – CPU-I/O burst cycle, CPU scheduler, Preemptive scheduling, Dispatcher 4.2 Scheduling Criteria 4.3 Scheduling Algorithms – FCFS, SJF, Priority scheduling, Round-robin scheduling, Multiple queue scheduling, Multilevel feedback queue scheduling | 06 |
| Unit 5 | 5 Process Synchronization 5.1 Background – Problems with Concurrency , Race Condition 5.2 Critical Section Problem – Peterson’s Solution(for two process) 5.3 Semaphores: Usage, Implementation 5.4 Classic Problems of Synchronization – Producer Consumer problem, Reader Writer problem, Dining Philosopher Problem | 05 |
| Unit 6 | 6 Deadlocks 6.1 System Model 6.2 Deadlock Characterization – Necessary Conditions, Resource Allocation Graph 6.3 Deadlock Prevention 6.4 Deadlock Avoidance - Safe State, Resource Allocation Graph Algorithm, Banker’s Algorithm 6.5 Deadlock Detection 6.6 Recovery From Deadlock – Process Termination, Resource Preemption | 08 |
| Unit 7 | 7 Memory Management 7.1 Background – Basic Hardware, Address Binding, Logical Versus Physical Address Space, Dynamic Loading, Dynamic Linking and Shared Libraries, Overlays 7.2 Swapping 7.3 Contiguous Memory Allocation – Memory Mapping and Protection, Memory Allocation, Fragmentation 7.4 Paging – Basic Method, Hardware Support, Protection, Shared | 12 |

| | | |
|---------------|---|-----------|
| | <p>Pages</p> <p>7.5 Segmentation – Basic Concept, Hardware</p> <p>7.6 Virtual Memory Management – Background, Demand Paging</p> <p>7.7 Page Replacement Algorithms – FIFO, OPT, LRU, Second Chance Page Replacement, LFU, MFU.</p> <p>7.8 Thrashing – Cause of thrashing , Working-set Model</p> | |
| Unit 8 | <p>8 File System</p> <p>8.1 File concept</p> <p>8.2 Access Methods – Sequential, Direct, Other access methods</p> <p>8.3 Directory and Disk Structure – Storage structure, Directory overview, Single level directory, Two level directory, Tree structure directory, Acyclic graph directory, General graph directory</p> <p>8.4 Allocation Methods – Contiguous allocation, Linked allocation, Indexed allocation</p> <p>8.5 Free Space Management – Bit vector, Linked list, Grouping, Counting, Space maps</p> | 07 |
| Unit 9 | <p>9 I/O Systems and Disk Management</p> <p>9.1 I/O Hardware - polling, interrupts, DMA</p> <p>9.2 Application I/O Interface - block and character devices, network devices, clocks and timers, blocking and non blocking I/O</p> <p>9.3 Kernel I/O subsystems - (I/O scheduling, buffering, caching, spooling and device reservation, error handling)</p> <p>9.4 Disk Structure</p> <p>9.5 Disk Scheduling – Disk Performance Parameters, Scheduling algorithms(FCFS, SSTF, SCAN,C-SCAN,LOOK,C-LOOK)</p> | 07 |

Reference Books:

1. “Operating System Concepts”, 9th Edition ,by Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, John Wiley & Sons (ASIA) Pvt. Ltd, 2013.
2. “Operating Systems: Internals and Design Principles”, 7th Edition, by William Stallings, Prentice Hall, 2011
3. “Modern Operating Systems”, 4th Edition, by Andrew S. Tanenbaum, Prentice Hall of India Pvt. Ltd, 2014.
4. “Operating Systems : Principles and Design” – Pabitra Pal Choudhary (PHI Learning Private Limited)
5. “An Introduction to Operating Systems, Concepts and Practice” by Pramod Chandra P. Bhatt , PHI, 2010
6. “Operating Systems: A Concept -based Approach”, 2nd Edition by D M Dhamdhare, Tata McGraw -Hill Education, 2007.

Websites:

<https://www.nptel.ac.in>

<http://www.scs.stanford.edu/17wi-cs140/>

<http://cnds.eecs.jacobs-university.de/courses/os-2016/slides.pdf>

<https://courses.cs.vt.edu/csonline/OS/Lessons/index.html>

Third Year B.C.A. (Science) Semester V
(To be implemented from Academic year 2017-18)

Course Code: BCA-505

Course Title: LAB I (Core Java)

Total Contact Hours: 48 hrs.

Total Credits: 04

Total Marks: 100

Note that these are only sample assignments. Teachers may conduct practicals by preparing similar types of examples

Java Sample Programs

1. Java Program to Print an Integer (Entered by the User)
2. Java Program to Add Two Integers
3. Java Program to Multiply two Floating Point Numbers
4. Java Program to Find ASCII Value of a character
5. Java Program to Compute Quotient and Remainder
6. Java Program to Swap Two Numbers
7. Java Program to Check Whether a Number is Even or Odd
8. Java Program to Check Whether an Alphabet is Vowel or Consonant
9. Java Program to Find the Largest Among Three Numbers
10. Java Program to Find all Roots of a Quadratic Equation
11. Java Program to Check Leap Year
12. Java Program to Check Whether a Number is Positive or Negative
13. Java Program to Check Whether a Character is Alphabet or Not
14. Java Program to Calculate the Sum of Natural Numbers
15. Java Program to Find Factorial of a Number
16. Java Program to Generate Multiplication Table
17. Java Program to Display Fibonacci Series
18. Java Program to Find GCD of two Numbers
19. Java Program to Find LCM of two Numbers

20. Java Program to Display Characters from A to Z using loop
21. Java Program to Count Number of Digits in an Integer
22. Java Program to Reverse a Number
23. Java Program to Calculate the Power of a Number
24. Java Program to Check Whether a Number is Palindrome or Not
25. Java Program to Check Whether a Number is Prime or Not
26. Java Program to Display Prime Numbers Between Two Intervals
27. Java Program to Check Armstrong Number
28. Java Program to Display Armstrong Number Between Two Intervals
29. Java Program to Display Prime Numbers Between Intervals Using Function
30. Java Program to Display Armstrong Numbers Between Intervals Using Function
31. Java Program to Display Factors of a Number
32. Java Program to Make a Simple Calculator Using switch...case
33. Java Program to Check Whether a Number can be Expressed as Sum of Two Prime Numbers
34. Java Program to Find the Sum of Natural Numbers using Recursion
35. Java Program to Find Factorial of a Number Using Recursion
36. Java Program to Find G.C.D Using Recursion
37. Java Program to Convert Binary Number to Decimal and vice-versa
38. Java Program to Convert Octal Number to Decimal and vice-versa
39. Java Program to Convert Binary Number to Octal and vice-versa
40. Java Program to Reverse a Sentence Using Recursion
41. Java Program to calculate the power using recursion

42. Java Program to Calculate Average Using Arrays
43. Java Program to Find Largest Element of an Array
44. Java Program to Calculate Standard Deviation
45. Java Program to Add Two Matrix Using Multi-dimensional Arrays
46. Java Program to Multiply to Matrix Using Multi-dimensional Arrays
47. Java Program to Multiply two Matrices by Passing Matrix to a Function
48. Java Program to Find Transpose of a Matrix
49. Java Program to Find the Frequency of Character in a String
50. Java Program to Count the Number of Vowels and Consonants in a Sentence
51. Java Program to Sort Elements in Lexicographical Order (Dictionary Order)
52. Java Program to Add Two Complex Numbers by Passing Class to a Function
53. Java Program to Calculate Difference Between Two Time Periods
54. Java Code To Create Pyramid and Pattern
55. Java Program to Remove All Whitespaces from a String
56. Java Program to Print an Array
57. Java Program to Convert String to Date
58. Java Program to Round a Number to n Decimal Places
59. Java Program to Concatenate Two Arrays
60. Java Program to Convert Character to String and Vice-Versa
61. Java Program to Check if An Array Contains a Given Value
62. Java Program to Check if a String is Empty or Null
63. Java Program to Get Current Date/Time
64. Java Program to Convert Milliseconds to Minutes and Seconds

65. Java Program to Add Two Dates
66. Java Program to Join Two Lists
67. Java Program to Convert List (ArrayList) to Array and Vice-Versa
68. Java Program to Get Current Working Directory
69. Java Program to Convert Map (HashMap) to List
70. Java Program to Convert Array to Set (HashSet) and Vice-Versa
71. Java Program to Convert Byte Array to Hexadecimal
72. Java Program to Create String from Contents of a File
73. Java Program to Append Text to an Existing File
74. Java Program to Convert a Stack Trace to a String
75. Java Program to Convert File to byte array and Vice-Versa
76. Java Program to Convert InputStream to String
77. Java Program to Convert OutputStream to String
78. Java Program to Lookup enum by String value
79. Java Program to Compare Strings
80. Java Program to Sort a Map By Values
81. Java Program to Sort ArrayList of Custom Objects By Property
82. Java Program to Check if a String is Numeric

Third Year B.C.A. (Science) Semester V
(To be implemented from Academic year 2017-18)

Course Code: BCA-506

Course Title: LAB II (Adv. Web Technology)

Total Contact Hours: 48 hrs.

Total Credits: 04

Total Marks: 100

Note that these are only sample assignments. Teachers may conduct practicals by preparing similar types of examples

PHP Slips for T Y BCA

- 1) Write class declarations and member function definitions for an **employee**(code, name, designation). Design derived classes as **emp_account**(account_no, joining_date) from employee and **emp_sal**(basic_pay, earnings, deduction) from emp_account.
Write a PHP Script to create 5 objects (pass details using __Construct () constructor) and Display details Employees who having Maximum and Minimum Salary.
- 2) Define an interface which has methods area(), volume(). Define constant PI. Create a class cylinder which implements this interface and calculate area and volume. (Use define())
- 3) Derive a class square from class Rectangle. Create one more class circle. Create an interface with only one method called area(). Implement this interface in all the classes. Include appropriate data members and constructors in all classes. Write a program to accept details of a square, circle and rectangle and display the area.
- 4) Create an abstract class Shape with methods calc_area() and calc_volume(). Derive three classes Sphere(radius) , Cone(radius, height) and Cylinder(radius, height), Calculate area and volume of all. (Use Method overriding).
- 5) Define a class Employee having private members – id, name, department, salary. Define parameterized constructors. Create a subclass called “Manager” with private member bonus. Create 6 objects of the Manager class and display the details of the manager having the maximum total salary (salary + bonus).
- 6) Write a PHP Script to create a super class **Vehicle** having members Company and price. Derive 2 different classes LightMotorVehicle (members – mileage) and HeavyMotorVehicle (members – capacity-in-tons). Define 5 Object of each subclass and display details in table format.
- 7) Write PHP script for the following: Define Class declarations and member function definitions for Student(rollno, name, academic_year). Design derived classes as Internal(marks, total), External(marks, total). Perform the following operations and show the results: Accept the details from the user and Show the result along with all details and total marks as addition of marks of Internal and External.
- 8) Write a script to keep track of number of times the web page has been accessed(use \$_COOKIE).

- 9) Create a login form with a username and password. Once the user logs in, the second form should be displayed to accept user details (name, city, phoneno). If the user doesn't enter information within a specified time limit, expire his session and give a warning otherwise Display Details(\$_SESSION).
- 10) Create a form to accept student information (name, class, address). Once the student information is accepted, accept marks in next form (Java, PHP, ST, IT, pract1, and project). Display the mark sheet for the student in the next form containing name, class, marks of the subject, total and percentage(Use \$_COOKIE).
- 11) Write a program to create a shopping mall. User must be allowed to do purchase from three pages. Each page should have a page total. The fourth page should display a bill, which consists of a page total of whatever the purchase has been done and print the total. (Use \$_SESSION).
- 12) Create a form to accept customer information (name, address, ph-no) (use Java Script to validate fields).Once the customer information is accepted, accept product information in the next form(Product name, qty, rate). Display the bill for the customer in the next form. Bill should contain the customer information and the information of the products entered.
- 13) Write a PHP script to accept username and password. If in the first three chances, username and password entered is correct, then display second form with well come message, otherwise display error message.
- 14) Create student registration form and display details in the next page. (Use sticky form concept).
- 15) Write a PHP Script to display Server information in table format (Use \$_SERVER).
- 16) Write a PHP Script to Upload the file and display its information.(use \$_FILES).
- 17) Write a PHP program to accept username and password from the user. Validate it against the login table in the database. If there is a mismatch between username and password, then, display the error message as —invalid user name and password, else display the message as —Login successfull on the browser.
- 18) Write a PHP program to implement Create, Read, Update and Display operations on Employee table with attributes(empno, empname, dateOfJoin, address, salary). **(Use Radio Buttons)**
- 19) Consider the following relational database:
Project (Pgroupno, ProjectTitle)
Student (Seat no, Name, Class, Pgroupno)

Write a PHP script to accept project title and display list of students those who are working in a particular project.

20) Consider the following entities and their relationships

Emp (emp_no, emp_name, address, phone, salary)

Dept (dept_no, dept_name, location)

Emp-Dept are related with one-many relationship

Create a RDB in 3NF for the above and solve following

Using above database write a PHP script which will

- a) Insert employee records in table .
- b) Print a salary statement in the format given below, for a given department. (Accept department name from the user).

| Maximum Salary | Minimum Salary | Sum Salary |
|----------------|----------------|------------|
| | | |

21) Consider the following entities and their relationships

Doctor (doc_no, doc_name, address, city, area)

Hospital (hosp_no, hosp_name, hosp_city)

Doctor and Hospital are related with many-many relationship

Create a RDB in 3 NF for the above and solve following

Using above database, write a PHP script which accepts hospital name and print information about doctors visiting / working in that hospital in tabular format.

22) Consider the following entities and their relationships

Movie (movie_no, movie_name, release_year)

Actor (actor_no, name)

Relationship between movie and actor is many – many with attribute rate in Rs. Create a RDB in 3 NF for the above and solve following Using above database, write PHP scripts for the following:

(Hint: Create HTML form having two radio buttons)

- a) Accept actor name and display the names of the movies in which he has acted.
- b) Insert new movie information.

23) Consider the following entities and their relationships

BillMaster (billno, custname, billdate)

BillDetails (itemname, qty, rate, discount)

BillMaster and BillDetails are related with one-to-many relationship.

Create a RDB in 3 NF for the above and solve following

Write PHP script to print the bill in following format Accept the Bill number from user.

BillNo :

BillDate :

Customer Name :

| SrNo | Particular | Quantity | Rate | Discount | Total |
|------|------------|----------|------|----------|-------|
| | | | | | |
| | | | | | |

Gross Amount :

24) Write a script to create XML file named “Rajashree.xml”

The element details of “Rajashree .xml” are:

```
<Rajashree Productions>
  <Movie>
    <MovieName>.....</MovieName>
    <Actor Name>.....</ActorName>
    <ReleaseYear>.....</ReleaseYear>
  </Movie>
</RajashreeProductions>
```

Store details of at least 5 movies which got released during 1990-2015.

25) Write a PHP script to generate an XML in the following format

```
<?xml version = “1.0” ?>
<BookStore>
  <Books>
    <PHP>
      <title>Programming PHP</title>
      <publication>O'RELLY</publication>
    </PHP>
    <PHP>
      <title>Beginners PHP</title>
      <publication>WROX</publication>
    </PHP>
  </Books>
</BookStore>
```

26) Create a XML file which gives details of books available in “ABC Bookstore” from following categories.

- 1) Technical
- 2) Cooking
- 3) Yoga

and elements in each category are in the following format

```
<Book>
  <Book_PubYear>-----</Book_PubYear>
  <Book_Title> -----</Book_Title>
  <Book_Author> -----</Book_Author>
</Book>
```

Save the file as “Book.xml”

Create an application that reads “Book.xml” file into simple XML object. Display attributes and elements.

(Hint: Use simple_xml_load_file() function)

- 27) Write a script to solve following questions (Use “Book.xml” file)
- Create a DomDocument object and load this XML file.
 - Get the output of this Document to the browser
 - Save this [. XML] document in another format i.e. in [.doc]
 - Write a XML program to print the names of the books available in “Book.xml” file.

- 28) Write a script to create “cricket.xml” file with multiple elements as shown below:

```
<CricketTeam>
  <Team country=”India”>
    <player>____</player>
    <runs>_____</runs>
    <wicket>____</wicket>
  </Team>
</CricketTeam>
```

Write a script to add multiple elements in “cricket.xml” file of category, country=”Australia”.

- 29) Write a PHP script to accept an .XML file which should comprise the following:

```
<cricket>
  <player>abc</player>
  <runs>1000</runs>
  <wickets>50</wickets>
  <noofnotout>10</noofnotout>
</cricket>
```

For at least 5 players. Display the details of players who have scored more than 1000 runs and at least 50 wickets.

- 30) Link this “Rajashree.xml” file to the CSS style sheet and get well formatted output as given below.

```
a)MovieName :
    Color: black,
    Font-family: Copperplate Gothic Light;
    Font Size: 16 pts;
    Font:Bold;
b)ActorName and ReleaseYear:
    Color: Red;
    Font-family: Bodoni MT;
    Font Size: 12 pts;
    Font: Bold
```

- 31) Write a php script using AJAX concept, to give Hint to user when he/she type city name in the text field.
- 32) Write a PHP script using AJAX concept, to check user name and password are valid or Invalid (use database to store user name and password).
- 33) Write a PHP script using AJAX concept, to develop user-friendly and interactive search engine.

Third Year B.C.A. (Under Science) Semester V

Course Code: BCA 507

Course Title: Soft Computing

Total Contact Hours: 24 hrs.
(30 Lectures)

Total Credits: 02

Total Marks: 50

Teaching Scheme: Theory- 03 Lect./ Week

Pre-requisites :

Knowledge of Set Theory

Course Objectives:

1. To learn the concept of soft computing.
2. Understand different soft computing techniques like Genetic Algorithms, Fuzzy Logic , Neural Networks and their combination.

| Unit No. | Contents | No. of Lectures |
|---------------|---|-----------------|
| Unit 1 | Introduction to Soft Computing 1.1 History of Soft Computing 1.2 Brief Introduction to Neural Networks, Genetic Algorithms and Fuzzy Systems 1.3 Applications of Soft Computing | 04 |
| Unit 2 | Fundamentals of Fuzzy Systems 2.1 Fuzzy sets: Basic Definition and Terminology 2.2 Member Functions 2.3 Fuzzy Logic and Relations 2.4 Extension Principle and its problems | 09 |
| Unit 3 | Fundamentals of Evolutionary Computing 3.1 Evolutionary Algorithms 3.2 Encoding 3.3 Operators of genetic Algorithms and its problems | 07 |
| Unit 4 | Fundamental of Neural Network 4.1 Introduction 4.2 Model of Artificial Neuron 4.3 Architectures 4.4 Learning Methods (Supervised and Unsupervised) 4.5 Perceptron and Back-propagation | 10 |
| | | 30 |

Reference Books:

References

1. Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications, S. Rajasekaran, G. A. Vijayalakshami, PHI.
2. Chin Teng Lin, C. S. George Lee, Neuro-Fuzzy Systems, PHI

3. Timothy Ross, Fuzzy Logic with Engineering Applications, TMH
4. Introduction to soft Computing, Eva Volna, 1st Edition, ISBN 978-87-403-0573-09
5. Kishan Mehrotra, Elements of Artificial Neural Network, MIT Press
6. E. Goldberg, Genetic Algorithms: Search and Optimization, Addison-Wesley
7. S.N. Sivanandan and S.N. Deepa, Principles of Soft Computing, Wiley India, 2007. ISBN: 10: 81-265-1075-7.
8. S. Rajasekaran and G.A.V.Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms, PHI, 2003.
9. J.S.R.Jang, C.T.Sun and E.Mizutani, Neuro-Fuzzy and Soft Computing, PHI, 2004, Pearson Education.