Course Type: - Major   **Semester**: - 4th

Paper Title: - **OBJECT ORIENTED PROGRAMMING THROUGH JAVA**   **Paper Code**:- CAPC1423M

Credit Weightage: - THEORY -04; PRACTICALS- 02   **Batch**: - 2023

Course Objective:
- To understand the basic object-oriented programming concepts and apply them in problem solving.
- To illustrate inheritance concepts for reusing the program.
- To demonstrate multitasking by using multiple threads and event handling.
- To understand the basics of java console and GUI based programming.

Course Outcomes:
- Demonstrate the behavior of programs involving the basic programming constructs like control structures, constructors, string handling and garbage collection.
- Demonstrate the implementation of inheritance (multilevel, hierarchical and multiple) by using extend and implement keywords.
- Use multithreading concepts to develop inter process communication.
- Understand the process of graphical user interface design and implementation using AWT or swings.

UNIT – I

OBJECT ORIENTED THINKING AND JAVA BASICS- Need for oop paradigm, summary of oop concepts, History of Java, Java buzzwords, data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, using final with variables, garbage collection, overloading methods and constructors, recursion, nested and inner classes, exploring string class.

UNIT – II

INHERITANCE, PACKAGES AND INTERFACES – Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance specialization, specification, construction, extension, limitation, combination, benefits of inheritance. Member access rules, super uses, using final with inheritance, polymorphism- method overriding, abstract classes, the Object class.
Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring java.io.

UNIT – III

EXCEPTION HANDLING AND MULTITHREADING Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes.String handling,Exploring java.util. Differences between multi-threading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, interthread communication, thread groups.

UNIT – IV

EVENT HANDLING: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes.
SWING: Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing-JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class,Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables. Handling menus, graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

TEXT & REFERENCES:

1. Java the complete reference, 7th edition, Herbert schildt, Tata McGraw Hill.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson education.
3. Core Java 2, Vol 1, Fundamentals, Cay.S.Horstmann and Gary Cornell, 8th Edition, Pearson Education.
4. Programming with Java, E. Balagurusamy, Tata McGraw Hill.
5. Java: How to Program P.J.Deitel and H.M.Deitel, PHI.

LAB. - Object Oriented Programming through Java (CAPC1423M)

LIST OF Programs:

1. Write a Java program to print "Hello, World!" on the console.
2. Create a program that takes two numbers as input and displays their sum, difference, product, and quotient.
3. Implement a program to check whether a given number is prime or not.
4. Develop a Java program to calculate the factorial of a given number using both iterative and recursive approaches.
5. Create a class representing a "Student" with attributes like name, roll number, and age. Include methods to display student details.
6. Design a "Rectangle" class with attributes length and width. Implement methods to calculate area and perimeter.
7. Develop a "BankAccount" class with attributes account number and balance. Include methods to deposit, withdraw, and display the balance.
8. Create a superclass "Shape" with attributes color and area. Derive subclasses like "Circle," "Square," and "Triangle" from the "Shape" class. Implement methods to calculate the area for each shape.
9. Design a "Vehicle" class with methods like start (), stop (), and accelerate (). Extend this class to create "Car" and "Motorcycle" subclasses with additional methods specific to each vehicle type.
10. Create an abstract class "Animal" with abstract methods like "sound ()" and "eat ()". Implement subclasses like "Dog," "Cat," and "Bird" with specific sound and eat behaviours.
11. Design an interface "Playable" with a method "playSound()". Implement classes like "Guitar," "Piano," and "Drums" that implement the "Playable" interface.
12. Create a program that takes user input for two numbers and performs division. Handle exceptions for division by zero and invalid input.
13. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in selected colour. Initially, there is no message shown.
14. Implement a program that reads data from a file and performs some calculations. Handle file-related exceptions like file not found or I/O errors.
15. Implement a program to store a list of names in an Array-List and perform operations like add, remove, and search for a name.
16. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
17. Create a HashMap to store student names and their corresponding marks. Display the student with the highest mark
18. Create a simple calculator GUI with buttons for numbers and operations (+, -, *, /).

Course Type: - Major	Semester: - 4th
Paper Title: - COMPUTER ORGANIZATION & ARCHITECTURE	Paper Code: - CAPC2423M
Credit Weightage: - THEORY -04; PRACTICALS- 02	Batch: - 2023

Course Objective:

- To introduce principles of computer organization and the basic architectural concepts.
- To familiarize with programming of a simple digital computer, simple register transfer language to specify various computer operations.
- Topics include computer arithmetic, instruction set design, micro programmed control unit, pipelining and vector processing, memory organization and I/O systems, and multiprocessors.

Course Outcomes:

- Understand the basics of instruction sets and their impact on processor design.
- Demonstrate an understanding of the design of the functional units of a digital computer system.
- Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
- Recognize and manipulate representations of numbers stored in digital computers.

UNIT – I

DIGITAL COMPUTERS: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture. Register Transfer Language and Micro operations: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit. Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

UNIT – II

MICRO PROGRAMMED CONTROL: Control memory, Address sequencing, micro program example, design of control unit. Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

UNIT – III

DATA REPRESENTATION: Data types, Complements, Fixed Point Representation, Floating Point Representation. Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating point Arithmetic operations.

UNIT – IV

INPUT-OUTPUT ORGANIZATION: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access. Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.
Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics. Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.

TEXT & REFERENCES:

1. Computer System Architecture – M. Morris Mano, Third Edition, Pearson/PHI.
2. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, McGraw-Hill.
3. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
4. Computer Architecture and Organization J.P. Hayes, McGraw-Hill.
5. Structured Computer Organization – Andrew S. Tanenbaum, PHI/Pearson.

LAB WORK - Computer Organization & Architecture (CAPC2423M)

<u>**LIST OF EXPERIMENTS:**</u>

1. Simulations of Control Unit......

Course Type: - Major
Paper Title: - DATA COMMUNICATION AND NETWORKS
Credit Weightage: - THEORY -04; PRACTICALS- 02

Semester: - 4th
Paper Code: - CAPC3423M
Batch: - 2023

Course Objective:

- Equip the students with a general overview of the concepts and fundamentals of computer networks.
- Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.

Course Outcomes:

- Gain the knowledge of the basic computer network technology.
- Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
- Obtain the skills of subnetting and routing mechanisms.
- Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation.

UNIT – I

Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.

Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless Transmission and transmission Impairments.Data link layer: Design issues, framing, Error detection and correction.

UNIT – II

Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channels.

Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols.

Medium Access sub layer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.

UNIT – III

Network Layer: Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, the Network layer in the internet.

UNIT – IV

Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols. Application Layer –Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.

TEXT & REFERENCES:

1. Data Communications and Networking – Behrouz A. Forouzan. Third, Tata McGraw Hill.
2. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI.
3. Computer Networking: A Top-Down Approach, 6th Edition, Pearson.

LAB. DATA COMMUNICATION AND NETWORKS (CAPC3423M)

LIST OF EXPERIMENTS:

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijsktra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting technique used in buffers.
10. Perform the following using NS2 Simulator
    I. NS2 Simulator-Introduction
    II. Simulate to Find the Number of Packets Dropped
    III. Simulate to Find the Number of Packets Dropped by TCP/UDP
    IV. Simulate to Find the Number of Packets Dropped due to Congestion
    V. Simulate to Compare Data Rate& Throughput.
    VI. Simulate to Plot Congestion for Different Source/Destination
    VII. Simulate to Determine the Performance with respect to Transmission of Packets ……

Course Type: - Minor   **Semester**: - 4th
Paper Title: - **INTRODUCTION TO DATABASE SYSTEMS**   **Paper Code**: - ACPC1423N

Credit Weightage: - THEORY -04; PRACTICALS- 02   **Batch**: - 2023

Course Objective:
- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To familiarize with data models, database design, relational model, relational algebra, transaction control, concurrency control.

Course Outcomes:
- Gain knowledge of fundamentals of DBMS, database design and normal forms.
- Master the basics of SQL for retrieval and management of data.
- Be acquainted with the basics of transaction processing and concurrency control.

UNIT – I

DATABASE SYSTEM APPLICATIONS: A Historical Perspective, File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS.
INTRODUCTION TO DATABASE DESIGN: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model.

UNIT – II

INTRODUCTION TO THE RELATIONAL MODEL: Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical data base design, introduction to views, destroying/altering tables and views. Relational Algebra, Tuple relational Calculus, Domain relational calculus.

UNIT – III

SQL: Queries, Constraints, Triggers: form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active data bases. SCHEMA REFINEMENT: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multi-valued dependencies.

UNIT – IV

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions..

TEXT & REFERENCES:
1. Database System Concepts, Silberschatz, Korth, Mc Graw Hill, 7th Edition.
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill.
3. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education.
4. Introduction to Database Systems, C. J. Date, Pearson Education.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. SQL, PL/SQL – The Programming Language of Oracle, Ivan Bayross, BPB Publications.

LAB WORK - INTRODUCTION TO DATABASE SYSTEMS (ACPC1423N)

LIST OF IMPLEMENTATIONS/PROGRAMES:

Implement the following on a given application:

1. Concept design with E-R Model.

2. Relational Model.

3. Normalization.

4. Practicing DDL commands.

5. Practicing DML commands.

6. Querying (using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.)

7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.

8. Triggers (Creation of insert trigger, delete trigger, update trigger).

9. Stored Procedures.

10. Usage of Cursors.