West Bengal State Council of Technical &
Vocational Education and Skill
Development
(Technical Education Division)



Syllabus

of

Diploma in Computer Science and
Technology [CST], Computer Science and
Engineering [CSE],
Computer Software Technology [CSWT] &
Information Technology [IT]

Part-II (4th Semester)

Revised 2022

## Semester IV

CURRICULUM STRUCTURE FOR PART-II (SEMESTER 4) OF THE FULL-TIME DIPLOMA COURSE in Computer Science and Technology [CST], Computer Science and Engineering [CSE], Computer Software Technology [CSWT] & Information Technology [IT]

| Sl. No | Category | Code No. | Course Title | Hours per week | | | Total contact hrs/ week | Credits |
|---|---|---|---|---|---|---|---|---|
| | | | | L | T | P | | |
| 1. | Program core course | COPC202 | Operating Systems | 2 | 1 | 0 | 3 | 3 |
| 2. | Program core course | COPC204 | Introduction to DBMS | 2 | 1 | 0 | 3 | 3 |
| 3. | Program core course | COPC206 | Computer Networks | 2 | 1 | 0 | 3 | 3 |
| 4. | Program core course | COPC208 | SSAD/Software Engineering | 3 | 0 | 0 | 3 | 3 |
| 5. | Program core course | COPC210 | Object Oriented Programming Using Java. | 3 | 0 | 0 | 3 | 3 |
| 7. | Minor Project | Proj.202 | Minor Project | 0 | 0 | 4 | 4 | 2 |
| 8. | Program core course | COPC212 | Operating Systems Lab | 0 | 0 | 2 | 2 | 1 |
| 9. | Program core course | COPC214 | Introduction to DBMS Lab | 0 | 0 | 2 | 2 | 1 |
| 10. | Program core course | COPC216 | Computer Networks Lab | 0 | 0 | 2 | 2 | 1 |
| 11. | Program core course | COPC218 | Object Oriented Programming Lab using Java | 0 | 0 | 2 | 2 | 1 |
| | **Total Credits** | | | | | | | **21** |

| Course Title: OPERATING SYSTEMS | |
|---|---|
| Course Code | COPC 202 |
| Number of Credits :3 | 3 (L: 2, T: 1, P: 0) |
| Prerequisites | Basic knowledge of computer organization, Data structure andhow computer works |
| Course Category | PC |
| Course code: CST | **Semester: FOURTH** |
| Duration: 15 weeks | **Maximum Marks: 100** |
| **Teaching Scheme** | **Examination Scheme** |
| Theory: 3  hrs/week<br><br>Total Contact Hours: 45 Hours | Continuous Internal Assessment: 20 Marks Attendance: 10 Marks Viva/Presentation/Assignment/Quiz etc: 10 MarksEnd Semester Examination: 60 Marks |

## Course Objectives

- ➢ Students will learn how Operating System is Important for Computer System.
- ➢ To make aware of different types of Operating System and their services.
- ➢ To learn different process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- ➢ To know virtual memory concepts.
- ➢ To learn secondary memory management.

**Course Content:**

| Contents (Theory) | Hrs./Unit | Marks |
|---|---|---|
| **UNIT 1: Introduction to Operating System** | 5 | 8 |

- Overview of Operating System (OS),
- OS generations
- Operating system structure,
- OS Kernel,
- OS services and systems calls,
- System programs.

| | | |
|---|---|---|
| **UNIT 2: Process Management and Concurrency Control** | **20** | 18 |

- Process Concept, Process states, Process control, Threads,

- Uni-processor Scheduling: Preemptive, Non preemptive, Scheduling algorithms:FCFS, SJF, RR, Priority, Thread Scheduling, Real Time Scheduling.

- IPC: Concept, Shared-Memory and Message-Passing Systems.

- Concurrency: Principles of Concurrency, Mutual Exclusion: S/W approaches, H/W Support, Semaphores, pipes, Message Passing, signals, Monitors,

- Classical Problems of Synchronization: Readers-Writers, and Dining Philosopher problem.

- Deadlock: Principles of deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock detection.

| UNIT 3: Memory management | 10 | 18 |
|---|---|---|

- Memory Management requirements, Memory partitioning: Fixed and Variable Partitioning,Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Fragmentation, Swapping, and Paging. Segmentation, Demand paging
- Virtual Memory: Concepts, management of VM, Page Replacement Policies (FIFO, LRU,Optimal, Other Strategies), Thrashing.

| UNIT 4: I/O management & Disk scheduling: | 6 | 10 |
|---|---|---|

- I/O Devices, Organization of I/O functions, Operating System Design issues, I/O Buffering.
- Disk Scheduling (FCFS, SCAN, C-SCAN, SSTF), RAID, Disk Cache.

| UNIT 5: OS Security | 4 | 6 |
|---|---|---|

- OS Security: Goals of Protection, Principle of protection, Access matrix, Access Control,Access Rights.
- Program Threats, Network and system threats, Cryptography, Authentication.

## Course outcomes

- Understands the different services provided by Operating System at different level.
- They learn real life applications of Operating System in every field.
- Understands the use of different process scheduling algorithm andsynchronization techniques to avoid deadlock.
- They will learn different memory management techniques like paging, segmentation and demand paging etc.

## Reference Books

1. Operating System Concepts, Silberschatz and Galvin, Wiley India Limited
2. Operating Systems, S. Haldar and Alex A. Aravind. Pearson Education, India
3. Operating Systems, Internals and Design Principles, Stallings, Pearson Education, India
4. Operating Systems, Nutt,Chaki and Neogy, Pearson Education, India
5. Modern Operating Systems, Andrew S. Tanenbaum, Prentice Hall of India
6. Operating systems, Deitel & Deitel, Pearson Education, India
7. Operating Syatems, Sumitradevi and Banashree, SPD

# West Bengal State Council of Technical & Vocational Education and Skill Development (Technical Education Division)

| Unit No. | Unit Title | Group | Distribution of Theory Marks | | | |
|---|---|---|---|---|---|---|
| | | | R Level | U Level | A Level | Total |
| 1. | Introduction to Operating System | A | 3 | 3 | 2 | 8 |
| 2. | Process Management and Concurrency Control | A | 6 | 6 | 6 | 18 |
| 3. | Memory management | B | 6 | 6 | 6 | 18 |
| 4. | I/O management & Disk scheduling | C | 3 | 4 | 3 | 10 |
| 5. | OS Security | C | 2 | 2 | 2 | 6 |
| | Total | | **20** | **18** | **22** | **60** |

**Legends: R = Remember; U = Understand; A = Apply and above levels**
**(Bloom's revisedtaxonomy)**

| Course Title: Introduction to DBMS | |
|---|---|
| Course Code | COPC 204 |
| Number of Credits : 3 | |
| Prerequisites | Having fundamental knowledge of Computers and elementary mathematics. |
| Course Category | PC |
| Course code: CST | Semester: Fourth |
| Duration: 15 weeks | Maximum Marks: 100 |
| Teaching Scheme | Examination Scheme |
| Theory: 3 hrs/week Total Contact Hours: 45 Hours | Continuous Internal Assessment: 20 Marks Attendance: 10 Marks Viva/Presentation/Assignment/Quiz etc.: 10 Marks End Semester Examination: 60 Marks |

**Aim of the Course**

- ➢ To study the Structure of Database Management System.
- ➢ To study different data Modeling.
- ➢ To study Relational Data modeling, Algebra & Calculus.
- ➢ To study SQL.
- ➢ To study functional Dependency and Normalization techniques.

**Course Objectives**

To enable student, develop database driven applications using the capabilities provided by modern database management system software which involves
- ✓ Conceptual modeling.
- ✓ Relational database design
- ✓ And Database Query Languages.

| Course Content: | | |
|---|---|---|
| Contents (Theory) | Hrs./Unit | Marks |
| **UNIT 1: Database System Concepts & Architecture** | 6 | 5 |
| 1.1 Basic concepts of DBMS, Advantages of a DBMS over file processing system, Data Abstraction, Database Languages, Data Independence. 1.2 Components of a DBMS and overall ANSI / SPARC architecture of DBMS. 1.3 Roles and functionalities of DBA & Data Dictionary. | | |
| **UNIT 2: Data Modeling Technique** | 5 | 10 |
| 2.1. Data Models: Hierarchical Model, Network Model, Object Oriented Model, E-R Model. Comparative study between different models. 2.2. The Enhanced Entity-Relationship (EER) model: Specialization, Aggregation, Generalization. | | |
| **UNIT 3: Relational Data Model in Depth** | 12 | 15 |

3.1. Relational Model: Basic concepts, attributes and domains.

3.2. Keys Concept: Super, Candidate, Primary and Foreign key.

3.3. Integrity constraints: Implicit and Explicit Constraints. Domain, Key, Entity and Referential Integrity Constraints.

3.4. Security & Authorization: Secrecy, Integrity, Availability.

3.5. Relational Algebra and Calculus.

| UNIT 4: Structured Query Language (should be taught preferably in Laboratory) | 12 | 15 |
|---|---|---|

4.1. Introduction to SQL queries: Creating, Inserting, Updating and deleting tables.

4.2. Table creation applying constraints (Primary key, Check constraints etc.

4.3. Set operations & operators, Aggregate functions, string functions and date, time functions.

4.4. Null values, Nested sub queries, Complex queries, Join concepts.

4.5. PL/SQL Introduction: PL/SQL block structure, variables, SQL statements in PL/SQL, PL/SQL control Structures, Cursors, Triggers, Functions, Packages, procedures.

| UNIT 5: Functional Dependencies & Normalization | 10 | 15 |
|---|---|---|

5.1. Purpose of Normalization, Data redundancy and updating anomalies, Functional Dependencies and Decomposition.

5.2. Process of Normalization using 1NF, 2NF, 3NF, multivalued dependencies and BCNF.

## Course outcomes

Student should be able to

- design a database, database-based applications

- understand the critical role of database system in designing several information system-based software systems or applications.

- use a DBMS in commercial query-based applications.

## Reference Books

| Name of Authors | Title of the Book | Edition | Name of the publisher |
|---|---|---|---|
| Abraham Silberschatz, Henry F. Korth, S. Sudarshan | Database System Concept | 5th | TMH |
| Evan Bayross | Database concepts and Systems | 3rd | SPD |
| P.S. Gill | Database Management System | 2nd | Wiley |
| Date, Kannan & Swaminathan | An Introduction to Database Systems | | Pearson |
| Navathe | Fundamentals of Database System | | Pearson |
| Singh | Database Systems | | Pearson |
| | | | |

# West Bengal State Council of Technical & Vocational Education and Skill Development (Technical Education Division)

| Unit No. | Unit Title | Group | Distribution of Theory Marks | | | |
|---|---|---|---|---|---|---|
| | | | R Level | U Level | A Level | Total |
| 1. | Database System Concepts & Architecture | A | 2 | 3 | | 5 |
| 2. | Data Modeling Technique | A | | 4 | 6 | 10 |
| 3. | Relational Data Model in Depth | A | 6 | 3 | 6 | 15 |
| 4. | Structured Query Language (should be taught preferably in Laboratory) | B | 3 | 6 | 6 | 15 |
| 5. | Functional Dependencies & Normalization | C | 4 | 6 | 5 | 15 |
| | Total | | **15** | **22** | **23** | **60** |

| Course Title : | **Computer Networks** |
|---|---|
| **Course Code** | COPC206 |
| **Number of Credits :** | 3 (L: 2, T: 1, P: 0) |
| **Prerequisites** | Having fundamental knowledge of Computers andelementary mathematics. |
| **Course Category** | CST |
| **Course code : General** | Semester : Fourth |
| **Duration : 15 weeks** | Total Marks : 100 |
| **Teaching Scheme** | Examination Scheme |
| **Theory : - 2 hrs/week** | Continuous Internal Assessment: 20 Marks |
| **Tutorial: - 1 hr/week** | Attendance, Assignment & Quiz : - 10, 5, 5 Marks respectively |
| **Total Contact Hours: 45 Hours** | End Semester Examination : 60 Marks |

**Aim:**

One of the major components of computer based information systems is computer networks. Through computer networks we can share hardware, Software, Processing, Data and Applications besides getting global connectivity for internet based communication and services.

For diploma students it is important to understand the function of computer networks and obtain requisite knowledge about hardware and software requirements of networks and acquire skills to establish a network using necessary hardware & software tools and configure various services over it. The objectives of this course are to make students learn the technology of establishing, commissioning(making operational) and maintaining computer networks.

**Course Objectives:** Understand functioning of computer networks and popular networking protocols

**Course Content:**

| **Contents (Theory)** | Hrs./Unit | Marks |
|---|---|---|
| **UNIT 1:** Basics of ComputerNetwork | 4 | 4 |

1.1 Definition & history of networks

1.2 Usage of Computer Networks

1.3 Standard Organizations and Protocols

1.4 Line Configuration

1.5 Network Topology

1.6 Periodic Analog Signals (Sine Wave, Phase, Wavelength, Time and Frequency Domains, Composite Signals, Bandwidth)

1.6 Categories of network Based on scope Based on Connection

1.7  Performance ( Bandwidth, Throughput, Latency (Delay), Bandwidth-Delay Product, Jitter)

| | | |
|---|---|---|
| **UNIT 2:** The Reference Model for network communication | **3** | 6 |

2.1 OSI model & function of each Layer

2.2 TCP/ IP model

2.3 Connection oriented v/s Connectionless approach Comparison of OSI & TCP/IP Models

| | | |
|---|---|---|
| **UNIT 3:** Transmission  Media | **5** | **4** |

3.1 Types of Transmission Media

3.2 Guided Media: Twisted Pair, CoaxialCable, Fiber

3.3  Un Guided Media: Electromagneticspectrum, Radio Transmission, Microwave Transmission, Infrared Transmission, Satellite Communication.

| | | |
|---|---|---|
| **Unit 4:** Multiplexing and Switching | 4 | 6 |

4.1 Multiplexing (Frequency- Division, Web length- Division, Time Division)

4.2 Introduction of Switching (Three Methods of Switching, Switching and TCP/IP Layers)

| | | |
|---|---|---|
| **UNIT 5:** Data Link Layer | **7** | **8** |

5.1 Design issues, example protocols (Ethernet, WLAN, Bluetooth)

5.2 Link- Layer Addressing (Three type of Address, Address Resolution Protocol(ARP))

5.3 Error Detection and Correction (Block Coding, Cyclic Codes, Checksum, Forward Error Correction)

5.4 Data Link protocols (Simple protocols, Stop-and-wait Protocol, Piggybacking)

5.6 Media Access Control (Random Access, Controlled Access, Channelization)

| | | |
|---|---|---|
| **Unit 6:** Network Layer | **6** | **8** |

6.1 Network Layer services (Packetizing, Routing and Forwarding)

6.2 Network-Layer Performance (Delay, Throughput, packet Loss, Congestion Control)

6.3 IPV4 Address (Address Space, Classful and Classless addressing)

6.4 Routing Algorithms

| | | |
|---|---|---|
| **Unit 7:** Transport Layer | **7** | **8** |

7.1 Transport layer Services

7.2 Connectionless and connection-oriented Protocols

7.3 Transport layer Protocols ( Simple Protocol, Stop-and-Wait Protocol, Go-Back-N Protocol, Selective-Repeat Protocols, Bidirectional protocol: piggybacking)

| | | |
|---|---|---|
| **Unit 8:** Network devices | **4** | **8** |

8.1 Repeaters

8.2 Hubs

8.3 Switches

8.4 Routers

8.5 Access Points

8.6 Gateways

8.7 Bridges

8.8 Difference between Layer 2 andLayer 3 Switches.

8.9 Introduction of Network Management software

| UNIT 9: Application Layer | 5 | 8 |
|---|---|---|

9.1   Principles of Network Application

9.2   The Web and HTTP

9.3   File Transfer: FTP

9.4   Electronic Mail in the Internet (SMTP, Comparison with HTTP, Mail Message Format, Mail access Protocols)

9.5  DNS

**9.6**  Peer-to-Peer Applications (P2P File Distribution)

## Reference Book

**1. Computer Networking by Kurose, Ross, Pearson;**
**2. Data Communications and Networking by Forouzan, Mc Graw Hill ;**
**3. Computer Networks by Andrew S Tannebaum, Pearson;**
**4. Head First Networking, SPD;**
**5. Computer Networks, Bhushan Trivedi, Oxford University Press**
**6. Networks for Computer Scientists and Engineers, Youlu Zheng & Shakil Akhtar, Oxford University Press**
**7. Data & Computer Communication by Williams Stallings, Pearson;**

## Course outcomes:

1.   Understanding of computer networks, issues, limitations, options available.

2.   Understanding of the care that needs to be taken while developing applications designed towork over computer networks

3.   Able to configure basic LAN and connect computers to it.

| Unit No. | Unit Title | Group | Distribution of Theory Marks | | | |
|---|---|---|---|---|---|---|
| | | | R Level | U Level | A Level | Total Marks |
| I | Basics of Computer Network | A | 4 | | | 4 |
| II | The Reference Model for network communication | A | 2 | 4 | | 6 |
| III | Transmission Media | A | 4 | | | 4 |
| IV | Multiplexing and Switching | A | 2 | 4 | | 6 |
| V | Data Link Layer | B | | 2 | 6 | 8 |
| vi | Network Layer | B | 1 | 1 | 6 | 8 |
| vii | Transport Layer | C | 1 | 1 | 6 | 8 |
| viii | Network devices | C | 2 | 6 | | 8 |
| ix | Application Layer | C | | 6 | 2 | 8 |
| | **Total** | **3** | **16** | **24** | **20** | **60** |

**Legends:** R = Remembrance; U= Understanding; A= Application and above levels (Revised Bloom's taxonomy)

# West Bengal State Council of Technical & Vocational Education and Skill Development (Technical Education Division)

| Course Title : SOFTWARE ENGINEERING | |
|---|---|
| **Course Code** | **COPC 208** |
| **Number of Credits :3** | 3 (L: 3, T: 0, P: 0) |
| **Prerequisites** | NIL |
| **Course Category** | PC |
| **Course code : CST** | Semester : FOURTH |
| **Duration : 15 weeks** | Maximum Marks : 100 |
| **Teaching Scheme** | Examination Scheme |
| **Theory : - 3 hrs/week** | Continuous Internal Assessment : 20 Marks |
| **Tutorial:NIL** | Attendance-10 Marks |
| **Total Contact Hours:45 Hours** | Viva/Presentation/Assignment /Quiz etc : - 10 Marks |
| **Practical : NIL** | End Semester Examination : 60 Marks |
| **Aim:** | Develop basic concept of Software Development in Computer Science |

### Course Objectives:

1. To provide an idea of using various process models in the software industry according to given circumstances.

2. Analysis of Proper Requirement Specification and importance of its documentation.

3. Emphasize on structured approach to handle software development.

4. Understanding of proper software Testing modules and its implementations

5. Inculcate essential technology and software engineering knowledge and skills essential to build reasonably complex usable and maintainable software iteratively.

### Course Content:

| Contents (Theory) | Hrs | Marks |
|---|---|---|
| **UNIT 1: Introduction to Software Engineering** | **07** | **08** |

- The evolving Role of software & changing nature of software.
- Software Engineering –A layered Technology approach.
- A process framework & software project tracking & control.
- The Capability Maturity Model Integration technique.
- Process patterns, process Assessment, personal & Team Process models & Process Technology Theories.
- Process Models –Waterfall, Incremental, RAD, Prototype, Spiral

| UNIT 2: Software Development Activities | 10 | 13 |
|---|---|---|

- Software Engineering core principles, Communication, Planning, Modelling, Construction & Deployment principles.
- Requirements Engineering Tasks, Initiating the requirement process.
- Analysis approaches of software & preparation of Analysis model using Data Modelling Concepts, Object oriented Analysis, Flow oriented model, Class-Based model, Behavioural Model.
- Design approaches of software & preparation of design model using Design concepts, Design model, and pattern based design.

| UNIT 3 : Software Testing Basics | 10 | 13 |
|---|---|---|

- Software Testing Fundamentals.
- A Strategic approach to software testing.
- Test Strategies for conventional software, Unit Testing, Integration Testing, Regression testing, smoke testing.
- Validation testing using Alpha & beta testing, system testing using recovery, security, stress & performance testing.
- Black Box & White Box Testing.
- Debugging process strategies.

| UNIT 4: Project Management | 10 | 14 |
|---|---|---|

- The management spectrum – The people, The product, the process & the project.
- Project scheduling – Basic concepts, relationship between people & effort, effort distribution, defining a task for the software project, Defining a task network & scheduling of project.
- Risk Management – Reactive Vs Proactive risk strategies, software Risks, Risk Identification, Risk Projection & Risk refinement, monitoring & management.
- Change Management – SCM scenario, SCM repository & process.
- Formal method & clean room software development & management approach.

| UNIT 5: Software Quality Management& Estimation | 08 | 12 |
|---|---|---|

- Basic Quality Concepts.
- Software Quality Assurance
- Statistical software quality assurance,
- Six sigma strategy.
- Software Reliability
- The ISO 9000 quality standards
- McCall's quality factors.
- Observations on estimation
- The project Planning process, software scope &
- feasibility, Resources
- Decomposition Techniques
- COCOMO II model & the make / Buy design

**Reference Books**

1. "Fundamental of Software Engineering", Rajib Mall, PHI
2. "Software Engineering –A Practitioner's Approach*"- Roger S. Pressman, TMH*
3. "Software Engineering" Sommerville , Pearson
4. "An Integrated Approach to Software Engineering"-Pankaj Jalote, Narosa
5. "Software Engineering" Udit Agarwal, S.K.Kataria &Sons
6. "Software Engineering" Sajan Mathew, S. Chand

**Course outcomes:**

- Students can plan & develop the frame work of project.
- Compare various project process models & use in project planning
- Use the principles of communication, planning, modelling construction & deployment
- Apply testing strategies & methods on software projects.
- Compare various testing methods.
- Identify the duties & responsibilities of People, team leader & stakeholders while planning the
  software project.
- Schedule the project according to time, size, shape, utility & application
- Monitor & manage the risk during the design of software project.
- Use the parameters of software quality assurance
- Calculate the cost of software, using cost estimation models such as COCOMO II.

| Unit No. | Unit Title | Group | Distribution of Theory Marks | | | |
| --- | --- | --- | --- | --- | --- | --- |
| | | | R Level | U Level | A Level | Total |
| 1. | Introduction to Software Engineering | A | 4 | 4 | 0 | 08 |
| 2. | Software Development Activities | A | 4 | 4 | 5 | 13 |
| 3. | Software Testing Basics | B | 4 | 4 | 5 | 13 |
| 4. | Project Management | C | 4 | 4 | 6 | 14 |
| 5. | Software Quality Management& Estimation | C | 4 | 4 | 4 | 12 |
| | **Total** | | **20** | **20** | **20** | **60** |

**Legends:** R = Remember; U = Understand; A = Apply and above levels (Bloom's revised taxonomy)

| Course Title : | **Object Oriented Programming using Java** |
|---|---|
| **Course Code** | COPC210 |
| **Number of Credits :** | 3 (L: 3, T: 0, P: 0) |
| **Prerequisites** | Any prior knowledge of programming language. |
| **Course code : General** | Semester : Fourth |
| **Duration : 15 weeks** | Total Marks : 100 |
| **Teaching Scheme** | Examination Scheme |
| **Theory : - 3 hrs/week** | Internal Assessment:   10 Marks |
| **Tutorial: - 0 hr/week** | Attendance, Assignment & Quiz : - 10, 5, 5 Marks respectively |
| **Total Contact Hours:** | 45 Hours |
| | End Semester Examination : 60 Marks |

**Aim:**

Open source platforms play significant role in the corporate world and are gaining popularity because these are freeware and ease of access. Java is a simple, portable, distributive, robust, secure, dynamic, architecture neutral, object oriented programming language.  This technology allows the software designed and developed once for an idealized 'virtual machine' and run on various computing platforms. Companies of all sizes are using Java as the main programming platform to develop various applications/projects worldwide. The aim of this course is that student should learn platform independent object oriented programming and java as base language for advanced technology like three tier architecture applications, cloud computing and web development. Many commercial applications as well as developing mission critical applications are using Java Technologies. This necessitates the corporate sectors to hire highly skilled Java developers. So, after learning this course, student can float themselves as Java developer in the software industry as well this course works as foundationcourse for advance Java programming for the forthcoming semester.

**Course Objectives:**
- Design and implement classes and methods
- Understand and implement basic programming constructs
- Apply object-oriented features to real time entities
- Understand and implement the concept of reusability and extensibility
- Create packages and interfaces and used it in programs
- Design and implement multithreaded programs
- Manage errors and exceptions
- Create packages and interfaces and used it in programs
- Design and implement applet and graphics programming
- Make use of Data streams in programs
- Make use of Data streams in programs

| Course Content: | | |
|---|---|---|
| **Contents (Theory)** | Hrs./Unit | Marks |
| **UNIT 1:** Introduction to Java | 5 | 7 |

1.1 Basics of Java, Background/History of Java, Java and the Internet, Advantages of Java
1.2 Java Virtual Machine & Byte Code
1.3 Java Environment Setup
1.4 Java Program Structure
1.5 Procedure-Oriented vs. Object-Oriented Programming concept
1.6 Basics of OOP: Abstraction, Inheritance, Encapsulation, Classes, subclasses and super classes, Polymorphism and Overloading, message communication
1.7 Compiling and running a simple "Hello World" program: Setting Up Your Computer,Writing a Program, Compiling, Interpreting and Running the program, Common Errors

| **UNIT 2:** Building Blocks of theLanguage | **8** | 10 |
|---|---|---|

2.1 Primitive Data Types: Integers, FloatingPoint type, Characters, Booleans etc
2.2 User Defined Data Type
2.3 Identifiers & Literals
2.4 Declarations of constants & variables
2.5 Type Conversion and Casting
2.6 Scope of variables & default values ofvariables declared
2.7 Wrapper classes
2.8 Comment Syntax
2.9 Garbage Collection
2.10 Arrays of Primitive Data Types
2.11 Types of Arrays
2.12 Creation, concatenation and conversion of a string, changing case of string, character extraction, String Comparison, String Buffer
2.13 Different Operators: Arithmetic, Bitwise, Rational, Logical,Assignment, Conditional,
Ternary, Increment and Decrement, Mathematical Functions
2.14 Decision & Control Statements: Selection Statement (if, if...else, switch),Loops (while, do-while, for), Jump statements (break, continue, return & exit)

| **UNIT 3:** Object Oriented Programming Concepts | **8** | 14 |
|---|---|---|

3.1 Defining classes, fields and methods, creating objects, accessing rules, thiskeyword, static keyword, method overloading, final keyword.
3.2 Constructors: Default constructors, Parameterized constructors, Passing object as a parameter, constructor overloading.

| **UNIT 4:** Inheritance, Packages & Interfaces | **10** | 12 |
|---|---|---|

| | |
|---|---|
| 4.1 Basics of Inheritance, Types of inheritance: single, multiple, multilevel, hierarchical and hybrid inheritance, concepts of method overriding, extending class, super class, subclass, dynamic method dispatch & Object class | |
| 4.2 Creating package, importing package, access rules for packages, class hidingrules in a package. | |
| 4.3 Defining interface, inheritance on interfaces, implementing interface,multiple inheritance using interface. | |

| UNIT 5: Exception Handling & Multithreaded Programming | **10** | 12 |
|---|---|---|

5.1 Types of errors, exceptions, try catch statement, multiple catch blocks, throwand throws keywords, finally clause, usesof exceptions, user defined exceptions
5.2 Creating thread, extending Thread class, implementing Runnable interface, life cycle of a thread, Thread priority & thread synchronization, exception handing in threads.

| **Unit 6:** File Handling | **4** | 5 |
|---|---|---|

6.1 Stream classes, class hierarchy, useful I/Oclasses, creation of text file, reading and writing text files

| **Reference Book** | | |
|---|---|---|

1. Core Java - Vol 1, 11e by Horstmann, Pearson;
2. Java by Anita Seth & B.L. Juneja, OXFORD;
3. Core Java for Beginners: A Simplified Approach (Covers Java SE 13) by Shah and Shah, SPD;
4. Java How to Program: Early Objects, 11e by Deitel, Pearson;
5. Java: The Complete Reference, Herbert Schildt, Tata McGraw Hill;

**Course outcomes:**

The theory should be taught in such a manner that studentsare able to acquire different learning out comes in cognitive, psychomotor and affective domain to demonstrate following course outcomes.

    i. Explain object oriented programming concepts of java.
    ii. Comprehend building blocks of OOPs language, inheritance, package andinterfaces.
    iii. Identify exception handling methods.
    iv. Develop multithreading object oriented programs.
    v. Develop an object oriented program handling data file.

| Unit No. | Unit Title | Group | Distribution of Theory Marks | | | |
|---|---|---|---|---|---|---|
| | | | R Level | U Level | A Level | Total |
| 1. | Introduction to Java | A | 4 | 3 | 0 | 7 |
| 2. | Building blocks of the Language | A | 2 | 2 | 6 | 10 |

| 3. | Object Oriented Programming Concepts | A | 4 | 4 | 6 | 14 |
|----|--------------------------------------|---|----|----|----|----|
| 4. | Inheritance, Packages and Interfaces | B | 2 | 4 | 6 | 12 |
| 5. | Exception Handling, Multithreaded Programming | C | 4 | 4 | 4 | 12 |
| 6. | File Handling | C | 0 | 3 | 2 | 05 |
| | **Total** | | **16** | **20** | **24** | **60** |

**Legends:** R = Remember; U = Understand; A = Apply and above levels (Bloom's revised taxonomy)

| Course Title: Minor Project | |
|---|---|
| **Course Code** | **Project 202** |
| **Number of Credits** | **02 (L: 0, T: 0, P: 4)** |
| **Prerequisites** | Having fundamental knowledge of Computers and elementary mathematics. |
| **Course Category** | PC |
| **Course code: CST** | **Semester: Fourth** |
| **Duration: 15 weeks** | **Maximum Marks: 100** |
| **Teaching Scheme** | **Examination Scheme** |
| Practical: 4 hrs/week<br>Total Contact Hours: 44 Hours | Continuous Internal Assessment: 60Marks<br>External Assessment: 40 Marks |

**Aim of the Course**

- To develop technical skill.
- To make use of hardware in developing Software.
- Analysis of different type of case studies.
- Software package development.
- Industrial practices in installation and maintenance of computers and computer networks.
- Fabrication of computers.
- Fault diagnosis and testing of computers.
- Industrial practices in respect of documentation and fabrication.
- A variety of computers and peripherals in assembly organizations.

**Course Objectives**

- ✓ Work in Groups, Plan the work and coordinate the work.
- ✓ Develop leadership qualities.
- ✓ Develop Innovative ideas.
- ✓ Practically implement the acquired knowledge.
- ✓ Develop basic technical Skills by hands on experience.
- ✓ Write project report.
- ✓ Develop skills to use latest technology in Computer/Information Technology field.
- ✓ Analyze the different types of Case studies.
- ✓ Use effectively oral, written and visual communication
- ✓ Demonstrate skill and knowledge of current information and technological tools and techniques specific to the professional field of study.
- ✓ Identify, analyse and solve problems creatively through sustained critical investigation.

**Course Content:**

| Sr. No. | Topics for Practice (Any one) | Skill Area |
|---|---|---|
| 01 | Notes and Password Manager | JAVA, MySQL |
| **02** | Scientific Calculator Application | C /C++ / Python/ VB |

| 03 | Evaluation of academic performance of Student(S) | PHP / Java, MySQL |
|----|---|---|
| 04 | Interactive Website Development of an Institute / Organization | JAVA Script/ PHP, MySQL |
| 05 | Breakout Ball Game | JAVA AWT, Swing, Frame etc. |
| 06 | Chatbot Song Recommender | Python API Chatbot |
| 07 | 1) Creating the file Server of your own : You can access the directory of the machine on which the Server code is running from the client machine using Sockets. Explore 'File' class in Java and you could better do that. You can also download the files. All you have to do is write a code for copying (Copying a file over a network)<br><br>2) Removable device detection system in LAN : Well, this is something again very interesting that you can do. The class used is same and a bit of logic. Whenever any hosts in the LAN adds up a directory or connects a USB removable drive, the server gives a notification with the hosts IP-Address<br><br>3) LAN Based Examination System | JAVA Networking |

| Course outcomes |
|---|

After completing the course:

- To enable students to implement Project Planning in their Industrial In-plant Training Project work.

- To be capable of self-education and clearly understand the value of achieving Perfection in the respective Project work.

- Apply fundamental and disciplinary concepts and methods in ways appropriate to their areas of study.

| Reference Books |
|---|

| Name of Authors | Title of the Book | Edition | Name of the publisher |
|---|---|---|---|
| Verhas Peter | Java Projects | 2nd | Packt Publishing Limited |
| Vishal Kumar Shah | Amazing Python Projects for Beginners | | Notion Press |
| A. Adams, C. Campbell, A. Khan | Programming Books Bundle with Project | | Code Academy |
| Horstmann | Core Java - Vol 1 | 11e | Pearson |
| Horstmann | Core Java - Vol 2 | 11e | Pearson |

| Course Title: Operating Systems Lab | |
|---|---|
| **Course Code** | COPC212 |
| **Number of Credits :1** | 2hrs/week |
| **Prerequisites** | Knowledge on Programming language like c/c++ |
| **Course Category** | PC |
| **Course Code: CST** | **Semester: FOURTH** |
| **Duration: 15 weeks** | **Maximum Marks: 100** |
| **Teaching Scheme** | **Examination Scheme** |
| Laboratory: 2 hrs/week Total Contact Hours: 30 Hours | Marks Continuous Internal Assessment:60 MarksExternal Assessment: 40 |

| **Course Objectives:** | | |
|---|---|---|
| This Lab course is intended to practice and do experiment on concepts taught in theory class of 'Operating Systems' and gain insight into functioning of the OperatingSystems. | | |

| **Course Content:** | | |
|---|---|---|
| **Sr. No.** | Topics for Practice | |
| | | Skill Area |
| 1 | Revision practice of various commands like man, cp, mv, ln, rm, unlink, mkdir, rmdir, etc and many more that were learnt in IT Workshop course and later. | LINUX |
| 2 | Simulate the CPU scheduling algorithms - Round Robin, SJF, FCFS,priority | C/C++ |
| 3 | Simulate Bankers algorithm for Deadlock Avoidance and Prevention | C/C++ |
| 4 | Simulate all FIFO Page Replacement Algorithm using C program | C/C++ |
| 5 | Simulate all LRU Page Replacement Algorithms using C program | C/C++ |
| 6 | Simulate Paging Technique of Memory Management | C/C++ |
| 7 | Practice various commands/utilitiessuch as catnl, uniq, tee, pg,comm, cmp, diff, tr, tar, cpio, mount, umount, find, umask, ulimit, sort, grep, egrep,fgrep cut, paste, join, du, df , ps, who, etc and manymore. | Linux |

| **Course outcomes** |
|---|
| • Students should be able to demonstrate basic knowledge about Operating System, • Be able to apply OS concepts such as processes, memory and file systems to systemdesign, • Be able to configure OS in an efficient and secure manner, and become an advance user of operating system. |

| **Reference Books** |
|---|
| |

➢ Operating Systems- Design and Implementation, Tanenbaum and Woodhull, PearsonEducation India
➢ Operating Systems- a concept based approach, Kaur and Gupta, Wiley
➢ UNIX Concepts and Applications, Sumitabha Das, McGraw-Hill Education

| Course Title: Introduction to DBMS Lab | |
|---|---|
| **Course Code** | **COPC214** |
| **Number of Credits : 1** | **2 (L: 0, T: 0, P: 2)** |
| Prerequisites | Basic Operations on Computer |
| Course Category | PC |
| **Course Code: CST** | **Semester: Fourth** |
| **Duration: 15 weeks** | **Maximum Marks: 100** |
| **Teaching Scheme** | **Examination Scheme** |
| Laboratory: 2 hrs/week<br>Total Contact Hours: 30 Hours | Continuous Internal Assessment: 60Marks<br>External Assessment: 40 Marks |
| **Course Objectives:** | |

This Lab course is intended to practice whatever is taught in theory class of 'Introduction to DBMS'. A few sample case studies are listed with some suggested activities. More case studies may be addedto this list. You need to develop these case studies, apply all relevant concepts learnt in theory classas the course progress, identify activities/operations that may be performed on the database. It willbe a good idea to also use concepts learnt in the course on Software Engineering/SSAD. Use of any "open-source database software" is highly appreciated.

**Course Content:**

| Sr. No. | Topics for Practice | Skill Area |
|---|---|---|
| 01 | • Creating & Executing DDL in SQL.<br>• Creating & Executing Integrity constraints in SQL.<br>• Creating & Executing DML in SQL.<br>• Executing relational, logical and mathematical set operators using SQL.<br>• Executing group functions<br>• Executing string operators & string functions.<br>• Executing Date & Time functions.<br>• Executing Data Conversion functions.<br>• Executing DCL in SQL.<br>• Executing Sequences and synonyms in SQL. | Familiarization with simple queries in SQL. |
| 02 | • Program for declaring and using variables and constant using PL/SQL.<br>• Program using if then else in PL/SQL.<br>• Program using for loop & while loop in PL/SQL.<br>• Program using nested loop in Pl/SQL. | Familiarization with PL/SQL. |

| | | |
|---|---|---|
| 03 | Employee database – 'Create' employee table, 'Select' and display an employee matching a given condition, 'Delete' duplicate records, delete rows using triggers,insert and update records, find net salary, etc. | Sample case Studies |
| 04 | Students' Academic database consisting of relevant tables to handle simple queries with PL/SQL. | Sample case Studies |
| 05 | Inventory Management System database. Do related activities such as prepare E-R Model, Relational Model, do Normalization, Create Tables, Insert data, Delete Data, Query database, create stored procedures, etc. | Sample case Studies |
| 06 | Bus Operator (Roadways) – Do related activities such as prepare E-R Model, Relational Model, do Normalization, Create Tables, Insert data, Delete Data, Query database, create stored procedures, etc. | Sample case Studies |

## Course outcomes

After completing the course, the students will understand –

- How to design a database, database-based applications.

- How to use a DBMS in Commercial applications.

- The critical role of database in designing for database software systems or database applications.

## Reference Books

| Name of Authors | Title of the Book | Edition | Name of the publisher |
|---|---|---|---|
| Joel Murach | Murach's SQL and PL/SQL | 2ND | SPD |
| Allen | Introduction to Relational Databases and SQL programming. | | Wiley |
| Deshpande | SQL and PL/SQL for Oracle 11g | | Dreamtech |
| Dasgupta | Database Management System, Oracle. SQL and PLSQL | | PHI |
| Priyadarsini | Database management System | | Scitech |

| Course Title : Computer Networks Lab | |
|---|---|
| **Course Code** | COPC216 |
| **Number of Credits :** | 1 (L:, T: 0, P: 2) |
| **Prerequisites** | An understanding of the TCP/IP protocol and the its layer model is recommended. Basic knowledge of python (such as through Intro to Computer Science) is required. |
| **Course Category** | CST |
| | Semester : Fourth |
| **Duration : 15 weeks** | Total Marks : 100 |
| **Teaching Scheme** | **Examination Scheme** |
| **Laboratory: 2 hrs/week** **Total Contact Hours: 30 Hours** | Continuous Internal Assessment: 60 Marks External Assessment: 40 Marks |
| | |

**Aim:**

The practical should be carried out in such a manner that students are able to acquire different learning out comes in cognitive, psychomotor and affective domain to demonstrate following course outcomes.

  i.     Describe various protocols, models in networks.
  ii.    Explain operations of TCP, HTTP, and DNS.
  iii.   Illustrate use of Subnets, Ipv4 in computer networks.
  iv.    Design simple computer networks.
  v.     Establish and Commission simple computer networks
  vi.    Identify and solve network operational problems.

| S.No. | Topics for Practice |
|---|---|
| 1 | Showing various types of networking cables and connectors, identifying them clearly |
| 2 | Looking at specifications of cables and connectors of various companies on Internet, find out differences. |
| 3 | Making patch cords using different types of cables and connectors - crimping, splicing, etc |
| 4 | Demonstration of different type of cable testers, using them for testing patch cords pre-pared by the students in Lab and standard cables prepared by professionals |
| 5 | Configuring computing devices (PC, Laptop, Mobile, etc) for network, exploring different options and their impact – IP address, gateway, DNS, security options, etc |
| 6 | Showing various networking devices – NICs, Hub, Switch, Router, WiFi access point, etc. |
| 7 | Looking at specifications of various networking devices various companies on Internet, find out differences. |
| 8 | Network simulation tool (e.g. Cisco Packet Tracer) |
| 9 | Setting up a small wired LAN in the Lab |
| 10 | Setting up a small wireless LAN in the Lab |

## SUGGESTED LIST OF STUDENT ACTIVITIES

i. Identify type of Network in your Institute.
ii. Prepare a design of Network in your Institute
iii. Visit your Institute server room and various places where Racks and servers installed, identify various Network components, collect information about installation of necessary hardware and software.
iv. Visit any ISP in your area.
v. Prepare Charts of Network Topologies.
vi. Seminar presentations.

## Reference Books:

1. Cisco press books on CCNA
2. User manual of networking devices available in the lab
3. Wiki pages on networking devices

## Course outcomes:

1. Understanding of computer networks, issues, limitations, options available.
2. Able to configure basic small LAN and connect computers to it.

## List of Major Equipment/ Instrument with Broad Specifications

i. Computer systems
ii. Network Cable Cat 5/Cat 6.
iii. Crimping Tool
iv. UTP Cable Tester
v. Layer 2 Switch
vi. Wireless Access point and Wireless router
vii. Impacting Tool
viii. Network cable connectors
ix. Network Trainer Kit.

## Course Title : Object Oriented Programming Lab using Java

| | |
|---|---|
| **Course Code** | COPC218 |
| **Number of Credits :** | 1 (L:, T: 0, P: 2) |
| **Prerequisites** | Any prior knowledge of programming language. |
| **Course Category** | CST |
| **Course code : General** | Semester : Fourth |
| **Duration : 15 weeks** | Maximum Marks : 100 |
| **Teaching Scheme** | **Examination Scheme** |
| **Laboratory: 2 hrs/week Total Contact Hours: 30 Hours** | Continuous Internal Assessment: 60Marks External Assessment: 40 Marks |
| **Aim:** | Develop software applications using object oriented concept in a Java SDK environment |

### Course Objectives:
 1.Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
2. Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
3. Be aware of the important topics and principles of software development.
4. Have the ability to write a computer program to solve specified problems.
5. Be able to use the Java SDK environment to create, debug and run simple Java programs

The practical/exercises are designed to develop different types of skills of the competency. Following is the list of practical problems.

| Sr. No | Unit No. | Exercise/Practical (Outcomes in Psychomotor Domain) |
|---|---|---|
| 1 | 1 | Install JDK, write a simple "Hello World" or similar java program, compilation, debugging, executing using java compiler and interpreter. |
| 2 | 2 | Write a program in Java to generate first n prime numbers. |
| 3 | 2 | Write a program in Java to find maximum of three numbers using conditional operator |
| 4 | 2 | Write a program in Java to find second maximum of n numbers without using arrays |
| 5 | 2 | Write a program in Java to reverse the digits of a number using while loop |
| 6 | 2 | Write a program in Java to convert number into words & print it |
| 7 | 2 | Write programs in Java to use Wrapper class of each primitive data types |
| 8 | 2 | Write a program in Java to multiply two matrix |

| 9 | 3 | Write a static block which will be executed before main( ) method in a class. |
|---|---|---|
| 10 | 3 | Write a program in Java to demonstrate use of **this** keyword. Check whether **this** can access the private members of the class or not. |
| 11 | 3 | Write a program in Java to develop overloaded constructor. Alsodevelop the copy constructor to create a new object with the state of the existing object. |
| 12 | 3 | Write a program in Java to demonstrate the use of private constructorand also write a method which will count the number of instances created using default constructor only. |
| 13 | 3 | Write a program in Java to demonstrate the use of **'final'** keyword in the field declaration. How it is accessed using the objects. |
| 14 | 3 | Develop minimum 4 program based on variation in methods i.e. passing by value, passing by reference, returning values and returning objects from methods. |
| 15 | 4 | Write a program in Java to demonstrate single inheritance, multilevel inheritance and hierarchical inheritance. |
| 16 | 4 | Create a class to find out whether the given year is leap year or not. (Use inheritance for this program) |
| 17 | 4 | Write an application that illustrates how to access a hidden variable. Class **A** declares a static variable **x**. The class **B** extends **A** and declaresan instance variable **x**. **display( )** method in **B** displays both of these variables. |
| 18 | 4 | Write a program in Java in which a subclass constructor invokes the constructor of the super class and instantiate the values. |
| 19 | 4 | Write a program that illustrates interface inheritance. Interface **P12** inherits from both **P1** and **P2**. Each interface declares one constant and one method. The class **Q** implements **P12**. Instantiate **Q** and invoke each of its methods. Each method displays one of the constants. |
| 20 | 4 | Write an application that illustrates method overriding in the same package and different packages. Also demonstrate accessibility rules in inside and outside packages. |
| 21 | 4 | Describe **abstract** class called **Shape** which has three subclasses say **Triangle**, **Rectangle**, **Circle**. Define one method **area()**in the abstract class and override this **area()** in these three subclasses to calculate for specific object i.e. area() of Triangle subclass should calculate area of triangle etc. Same for **Rectangle** and **Circle** |
| 22 | 4 | Write a program in Java to demonstrate implementation of multiple inheritance using interfaces. |
| 23 | 4 | Write a program in Java to demonstrate use of final class. |
| 24 | 5 | Write a program in Java to develop user defined exception for 'Divide by Zero' error. |
| 25 | 5 | Write a program in Java to demonstrate multiple try block and multiple catch exception |
| 26 | 5 | Write an small application in Java to develop Banking Application in which user deposits the amount Rs 1000.00 and then start withdrawingof Rs 400.00, Rs 300.00 and it throws exception "Not Sufficient Fund" when user withdraws Rs. 500 thereafter. |

| 27 | 5 | Write a program that executes two threads. One thread displays "Thread1" every 2,000 milliseconds, and the other displays "Thread2" every 4,000 milliseconds. Create the threads by extending the **Thread** class |
|----|---|---|
| 28 | 5 | Write a program that executes two threads. One thread will print the even numbers and the another thread will print odd numbers from 1 to50. |
| 29 | 5 | Write a program in Java to demonstrate use of synchronization of threads when multiple threads are trying to update common variable. |
| 30 | 6 | Write a program in Java to create, write, modify, read operations on a Text file. |

## SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:
  i.   Study available small Java application on internet and reuse in your application
  ii.  Develop Java object oriented application programs
  iii. Present the application developed

**Reference Book:**

1. Practical Java Project for Beginners, B. M. Harwani, SPD;
2. Intro to Java Programming, Comprehensive Version, 10e, Liang, Pearson;

## Course Outcome

On completion of the course the student should be able to:

- Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.
- Read and make elementary modifications to Java programs that solve real-world problems.
- Validate input in a Java program.
- Identify and fix defects and common security issues in code.
- Document a Java program using Javadoc.
- Use a version control system to track source code in a project.

**(A) List of Major Equipment/Materials**

  i.   Computer System with latest configuration and memory
  ii.  Multimedia projector
  iii. Internet Access
  iv.  Access to library resources

**(B) List of Software/Learning Websites**

  i.   Java Development Kit: http://www.oracle.com/technetwork/java/javase/downloads/index.html
  ii.  http://docs.oracle.com/javase/specs/jls/se7/html/index.html
  iii. http://docs.oracle.com/javase/tutorial/java/index.html
  iv.  http://www.tutorialspoint.com/java/
  v.   http://www.learnjavaonline.org/

vi.   http://www.c4learn.com/javaprogramming/
vii.  http://www.learn-java-tutorial.com/
viii. http://www.tutorialspoint.com/javaexamples/