Shirpur Education Society's

R. C. Patel Institute of Technology, Shirpur

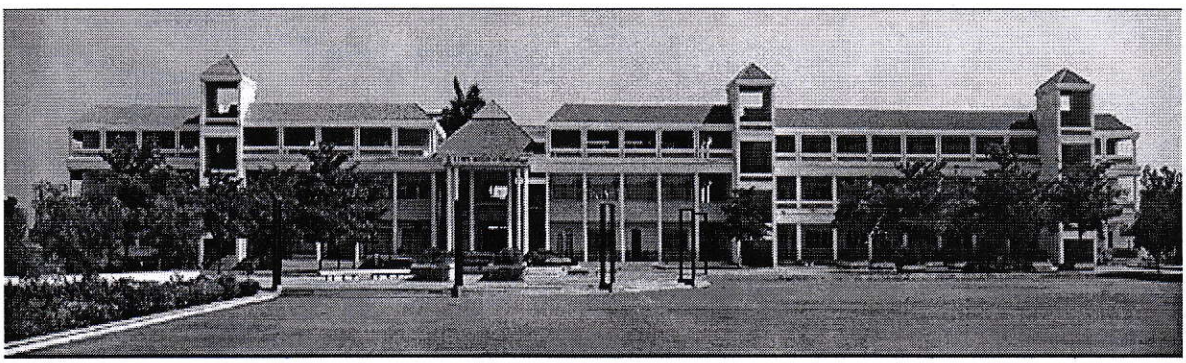
(An Autonomous Institute)

Structure and Syllabus

For

Minor in Computer Engineering

With effect from Year 2021-22



Shahada Road, Near Nimzari Naka, Shirpur, Maharashtra 425405
Ph: 02563 259802, Website: www.rcpit.ac.in

Introduction:

Minor aims for additional knowledge in any other branch for enhancement of employability.

1. Minor can be completed through MOOCs platform or classroom teaching mode.
2. Syllabus for Minor will be available through classroom teaching mode.
3. On successful completion of the requirements of Minor, the UG students shall be awarded a Certificate by Institute and shall be reflected on grade card.
4. Aspiring student shall register additional Theory courses and earn additional 18 (minimum) credits for each scheme to get Minor.

Learning Outcomes:

At the successful completion of this minor, UG students will be able to:

LO1: Design and develop computer based solutions to problems in their application domain.

LO2: Analyse a problem, and identify and define the computing requirements appropriate to its solution.

LO3: Apply principles of software engineering in the development of software systems.

Syllabus Structure & Scheme										
Course Code	Semester	Course Title	Teaching Scheme		Examination Scheme				Total	Credits
			L	P	Theory		Practical			
					ISE	ESE	ISE	ESE		
CMN201	SY Sem-III	Problem Solving through Programming in Java	3	2	-	-	40	60	100	4
CMN202	SY Sem-IV	Data Structures using Java	3	2	20	60	-	20	100	4
CMN301	TY Sem-V	DataBase Systems	3	2	20	60	-	20	100	4
CMN302	TY Sem-VI	Operating Systems	3	-	40	60	-	-	100	3
CMN401	TY Sem-VI	Software Engineering	3	-	40	60	-	-	100	3
		Total	15	6	120	240	40	100	500	18

Assessment Method:

Evaluation will be done by Internal Assessment tools and End Semester examinations.

Prepared by:
Prof. Mrs. S. P. Patil

Checked by:
Prof. Dr. Ms. Vandana M. Patil

Prof. Dr. Nitin N. Patil
BOS Chairman

Prof. S. P. Shukla
C.O.E.

Prof. Dr. P. J. Deore
Dean Academic/Dy. Director

Prof. Dr. J. B. Patil
Director

DEPT. OF COMPUTER ENGINEERING R.C.P.I.T. SHIRPUR OFFICE SEAL

Problem Solving through Programming in Java (CMN201)

Teaching Scheme	**Examination Scheme**
Lectures : 03 Hrs./week	Practical : ISE : 40 Marks
Practical : 02 Hrs./week	ESE : 60 Marks
Credits : 04	Total Marks : 100 Marks

Course Prerequisites: Basic Object-Oriented Concepts.

Course Objectives:

- Learn the object-oriented programming concepts.
- Study various java programming concepts like exception handling, packages etc.
- Apply algorithmic thinking to solve programming problems.
- Implement syntax rules in Java programs.

Course Outcomes:

After successful completion of the course, students will able to-

CO1: Apply fundamental programming constructs.

CO2: Illustrate the concept of packages, classes and objects.

CO3: Elaborate the concept of strings, arrays and vectors.

CO4: Implement the concept of inheritance and interfaces.

CO5: Implement the notion of exception handling.

Course Contents

Unit-I Introduction, Variables, Data Types, and Expressions 06 Hrs.

History of Java, Features of Java, How Java Works?, Types of Java Program, Edit, Compile, and Run Java Applications. Identifier Rules, Naming Variables, Constants (Final) and References, Primitive Data Types, Arithmetic Operators, Assignment Operators, Relational and Logical Operators.

Unit-II Program Control Flow 06 Hrs.

Sequence Structure, Selection Structure, Repetition Structure, Jump (Sequence) Structure.

Unit-III Methods 07 Hrs.

Java API and Package/Library Methods, User-defined Methods, Scope and Duration, Local and Field Variables, Pass-by-Value, Pass-by-Reference, Recursion, Overloading.

Unit-IV Arrays & Strings 07 Hrs.

Declaration and Allocation, Passing Arrays to Methods, Sorting, Searching, Multiple-subscripted, Vector, String.

Unit-V Object-Based Programming 07 Hrs.

Classes and Objects, Instance Variables, and Instance Methods, Member Access Modifiers: Public, Private, Protected, Package, Creating Packages, Constructors, Overloaded Constructors, Set (Mutator), Get (Access), and Predicate Methods, Final Instance Variables, Composition, Finalizers, Garbage Collection, Static Class Members, this Reference

Unit-VI Object-Oriented Programming 07 Hrs.

Inheritance, Super Class, Subclass, Polymorphism, Dynamic Method Binding, Abstract Class, Concrete Class, Inner Class Definition, Type-Wrapper class for Primitive Data Types, Interfaces.

List of Laboratory Assignments

1. Program on various ways to accept data through keyboard.
2. Program on branching, looping, labelled break and labelled continue.
3. Program to create class with members and methods, accept and display details for single object.
4. Program on constructor and constructor overloading.
5. Program on method overloading.
6. Program on passing object as argument and returning object.
7. Program on creating user defined package.
8. Program on 1D array.
9. Program on 2D array.
10. Program on String.
11. Program on Vector.
12. Program on single and multilevel inheritance.
13. Program on abstract class.
14. Program on interface demonstrating concept of multiple inheritance.
15. Program to demonstrate try, catch, throw, throws and finally. Also, implement user defined exception.
16. Mini Project based on syllabus concepts.

Text Books

1. Herbert Schildt, "JAVA: The Complete Reference", 9^{th} Edition, Oracle Press.
2. Sachin Malhotra and Saurabh Chaudhary, "Programming in Java", 2^{nd} Edition, Oxford University Press, 2010.

Reference Books

1. Ivor Horton, "Beginning JAVA", 7^{th} Edition, Wiley India.
2. Deitel and Deitel, "Java: How to Program", 8^{th} Edition,PHI.
3. "JAVA Programming", Black Book, 4^{th} Edition, Dreamtech Press.
4. "Learn to Master Java Programming", 11^{th} Edition, Staredu Solutions.

Data Structures using Java (CMN202)

Teaching Scheme	**Examination Scheme**
Lectures : 03 Hrs./week	Theory : ISE : 20 Marks
Practical : 02 Hrs./week	ESE : 60 Marks
Credits : 04	Practical ESE : 20 Marks
	Total Marks : 100 Marks

Course Prerequisites: Basic Object-Oriented Concepts in Java.

Course Objectives:

- To impart the basic concepts of Java.
- To explain fundamentals of data structures and their applications essential for programming/problem solving.
- To analyze Linear Data Structures: Stacks, Queues, Linked Lists.
- To analyze Non Linear Data Structures: Trees,Graphs.
- To assess appropriate data structure during program development/Problem Solving.
- To understand concepts about searching and sorting techniques.

Course Outcomes:

After successful completion of the course, students will able to-

CO1: Study, Understand and Implement basics of Java.

CO2: Implement linear data structures like Stacks and Queues.

CO3: Be familiar with implementation of linked data structures such as linked lists and binary trees.

CO4: Implement non-linear data structures like Trees and Graphs.

CO5: Ability to summarize searching and sorting techniques.

Course Contents

Unit-I Introduction to Java and basic OOP concepts — 08 Hrs.

How Java works?, Features of Java and object oriented concepts, Identifier rules, Naming variables, constants (final) and references, Primitive data types, Arithmetic Operators, Assignment Operators, Relational and Logical Operators, Control Structures, Loops, Methods, Arrays, Strings, Constructors.

Unit-II Stacks & Queues — 08 Hrs.

Stacks: Fundamentals of stacks, data structure and basic operations on stacks, applications of stacks: Function Call, Recursion, Tower of Hanoi, Infix, Postfix and Prefix notations, Inter conversions and evaluation of expressions, Multiple stacks.

Queues: Fundamentals of queues, Data structure and basic operations on linear queues, Disadvantages of linear queues, circular queues, double ended queues, priority queues, applications of queues.

Unit-III Linked Lists — 08 Hrs.

Linked lists: Types of linked lists, Representing linked list in java, Defining a Node in java, The Template Class , Circular Lists, Available Space Lists, Linked Stacks and Queues, Polynomials, Polynomial Representation.

Unit-IV Trees & Graphs — 08 Hrs.

Binary tree: Basic terminology, data structure and representation of binary tree, binary tree traversals, recursive and nonrecursive procedure for tree traversals, basic operations on binary tree (creation, insertion, deletion), Threaded binary trees, Concept of binary search trees, Basic operations on it - insertion, deletion and searching , Height balanced binary trees, LL, LR, RL, RR rotations.

Graphs: Basic terminologies, representation, DFS and BFS.

Unit-V Sorting and Searching — 08 Hrs.

Searching strategies: Linear and binary search algorithms, Bubble sort, Insertion sort, Selection sort, Quick sort, Merge sort, Best, Average and Worst case time complexities of each of the sorting and searching algorithms.

Hashing: Hashing functions, Collision, linear probing, deletion, clustering, rehashing, bucket and chaining, selection of good hash function.

List of Laboratory Assignments

1. Implementation of stack using array.
2. Implementation of queue using array.
3. Implementation of circular queue using array.
4. Implementation of single linked list and perform basic operations (insertion, deletion, display).
5. Create a hash table and handle the collisions using linear probing without replacement.
6. Implementation of array and string operations.
7. Implementation of insertion sort.
8. Implementation of selection sort.
9. Implementation of binary search.
10. Implementation of Depth First Search (DFS) and Breadth First Search (BFS).

Text Books

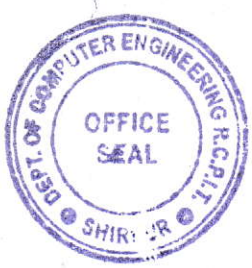
1. Horowitz and Sahni, "Fundamental of Data Structures", 4^{th} Edition, CSP, 1994, (Pascal, C , C++ or Generic version).
2. Carrano, F. M., "Data Abstraction and Problem Solving with C++", Benjamin Cummings, 1995.
3. Tenenbaum, A. M. Langsam, Augenstein, M. J., "Data Structures Using C++", 2^{nd} Edition, Prentice Hall, 1996.

Reference Books

1. Michael T. Goodrich, Roberto Tamassia, Norman Goldwasser, "Data Structures & Algorithms in Java", 6^{th} Edition, ISV (WSE).
2. Dr. Dheeraj Malhotra, Dr. Neha Malhotra, "Data Structures and Program Design using JAVA", Khanna Publishers.
3. Hemant Jain, "Problem Solving in Data Structures & Algorithms Using Java".
4. Thomas H. Corman, "Introduction to Algorithms", 3^{rd} Edition (The MIT Press).
5. Robert Sedgewick & Kevin Wayne, "Algorithms", 4^{th} Edition.
6. Steve S. Skiena, "The Algorithm Design Manual", 2^{nd} Edition,2008.
7. George T. Heineman, Gary Pollice, Stanley Selkow, "Algorithm in Nutshell", 2^{nd} Edition ThriftBooks-Dallas.

8. Manisha Suryavanshi, "Object Oriented Concepts and Programming in C++".

9. Mark Allen Weiss, "Data Structures and Algorithm Analysis", 3rd Edition.



Database Systems (CMN301)

Teaching Scheme	Examination Scheme
Lectures : 03 Hrs./week	Theory : ISE : 20 Marks
Practical : 02 Hrs./week	ESE : 60 Marks
Credits : 04	Practical ESE : 20 Marks
	Total Marks : 100 Marks

Course Prerequisites: Discrete Structures.

Course Objectives:

- To learn and practice data modelling using the entity-relationship and develop database designs.
- To understand the use of Structured Query Language (SQL) and learn SQL syntax.
- To apply normalization techniques to normalize the database.
- To understand the needs of database processing and learn techniques for controlling the consequences of concurrent data access.
- To introduce principles and foundations of distributed databases, design issues, query processing and optimization.

Course Outcomes:

After successful completion of the course, students will able to-

CO1: Illustrate the fundamentals of a database systems.

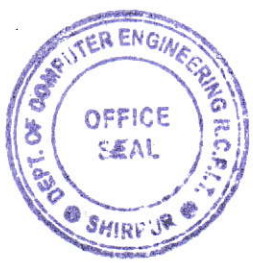
CO2: Design and draw ER and EER diagram for the real-life problem.

CO3: Build relational model from conceptual model and formulate relational algebra queries.

CO4: Design and query database using SQL.

CO5: Analyze and apply concepts of normalization to relational database design and explain the concept of transaction, concurrency and recovery.

CO6: Summarize the concepts of distributed database.



Course Contents

Unit-I Introduction to Database Concepts — 03 Hrs.

Introduction, Characteristics of Databases, File System v/s Database System, Users of Database System, Data Independence, DBMS System Architecture, Database Administrator.

Unit-II Entity–Relationship Data Model — 07 Hrs.

The Entity-Relationship (ER) Model: Entity Types: Weak and Strong Entity Sets, Entity Sets, Types of Attributes, Keys, Relationship Constraints: Cardinality and Participation. Extended Entity-Relationship (EER) Model: Generalization, Specialization and Aggregation.

Unit-III Relational Model and Relational Algebra — 08 Hrs.

Introduction to the Relational Model, Relational Schema and Concept of Keys, Mapping the ER and EER Model to the Relational Model. Relational Algebra: Unary and Set Operations, Relational Algebra Queries.

Unit-IV Structured Query Language (SQL) — 08 Hrs.

Overview of SQL, Data Definition Commands, Data Manipulation Commands, Data Control Commands, Transaction Control Commands.

Integrity Constraints: Key Constraints, Domain Constraints, Referential Integrity, Check Constraints, Set and String Operations, Aggregate Function, Group By Clause, Having Clause.

Views in SQL, Joins, Nested and Complex Queries.

Introduction to PL/SQL.

Unit-V Relational Database Design — 10 Hrs.

Pitfalls in Relational-Database Designs, Concept of Normalization, Functional Dependencies, First Normal Form, 2NF, 3NF, BCNF.

Transactions Management and Concurrency: Transaction Concept, Transaction States, ACID Properties, Concurrent Executions, Serializability–Conflict and View, Concurrency Control: Lock-Based, Timestamp-Based Protocols.

Recovery System: Introduction to Recovery System.

Unit-VI Distributed Database — 04 Hrs.

Introduction to Distributed Database, Features of DDBS, Design Issues in DDBS, Distributed Database Design Concept, Objectives, Data Fragmentation, Transparencies in Distributed Database Design.

List of Laboratory Assignments

1. Identify the case study and detailed statement of problem. Design an Entity-Relationship (ER)/ Extended Entity-Relationship (EER) Model.
2. Mapping ER/EER to Relational schema model.
3. Create and populate database using Data Definition Language (DDL) and Data Manipulation Language (DML) commands.
4. Apply various Integrity Constraints on given database.
5. To perform Simple queries, string manipulation operations on given database.
6. To perform Nested queries and Complex queries on given database.
7. To demonstrate use of different types of Join operations.
8. To demonstrate Views and Triggers.
9. Write a Procedures in PL/SQL to perform various operations on database.
10. Examine the consistency of database using concurrency control technique (Locks).
11. Case study on Fragmentation (PHF, DHF, VF, and HF) in DDBMS design.
12. Case study on recent databases and applications.

Text Books

1. Korth, Silberchatz, Sudarshan, "Database System Concepts", 6^{th} Edition, McGraw – Hill.
2. Elmasri and Navathe, "Fundamentals of Database Systems", 5^{th} Edition, Pearson Education.
3. Peter Rob and Carlos Coronel, "Database Systems Design, Implementation and Management", 5^{th} Edition, Thomson Learning.
4. Chhanda Ray, "Distributed Database System", Pearson Education India.
5. G. K. Gupta, "Database Management Systems", McGraw – Hill.

Reference Books

1. Dr. P. S. Deshpande, "SQL and PL/SQL for Oracle 10g", Black Book, Dreamtech Press.
2. Gillenson, Paulraj Ponniah, "Introduction to Database Management", Wiley Publication.
3. Sharaman Shah, Oracle for Professional, SPD.
4. Raghu Ramkrishnan and Johannes Gehrke, "Database Management Systems", 3^{rd} Edition, McGraw – Hill.
5. M. Tamer Ozsu, Patrick Valduriez, "Principles of Distributed Database", 2^{nd} Edition, Pearson Education India.

Operating Systems (CMN302)

Teaching Scheme	Examination Scheme
Lectures : 03 Hrs./week	Theory : ISE : 40 Marks
Credits : 03	ESE : 60 Marks
	Total Marks : 100 Marks

Course Prerequisites:

- Computer Organization.
- System Programming.

Course Objectives:

- To introduce basic concepts and functions of different operating systems.
- To understand the concept of process, thread and resource management.
- To understand the concepts of process synchronization and deadlock.
- To understand various memory, I/O and file management techniques.

Course Outcomes:

After successful completion of the course, students will able to-

CO1: Summarize basic functions of operating system.

CO2: Compare and evaluate process scheduling algorithms and IPC.

CO3: Illustrate various memory management techniques.

CO4: Explain and interpret file and I/O management techniques.

CO5: Discover functionalities of different operating systems.

Course Contents

Unit-I Introduction and Operating System Structures — 06 Hrs.

Definition, Operating System Objectives and Functions, Evolution of Operating System, Types of Operating system, Different OS Services and OS Components, System Calls and its Types, Operating System Structures.

Unit-II Processes and CPU Scheduling — 08 Hrs.

Process Concept, Process Scheduling, Operation on Process, Cooperating Processes, Scheduling: Concept, Objectives, Queuing Diagram, Types of Schedulers: Long Term Scheduler, Middle Term Scheduler, Short Term Scheduler, Scheduling Algorithm and Performance Evaluation (for Uniprocessor System): FCFS, SJF (Preemptive & Non Preemptive), Priority (Preemptive & Non Preemptive), Round Robin, MLQ with and without Feedback, IPC: Concept and Types, Threads.

Unit-III Process Synchronization — 06 Hrs.

The Critical-Section Problem, Critical Regions, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, and Monitors Synchronizations.

Unit-IV Deadlocks — 06 Hrs.

Systems Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

Unit-V Memory Management — 08 Hrs.

Memory Management: Basic Concept, Logical and Physical Address Map, Memory Allocation: Continuous Memory Allocation, Fixed and Variable Partition, Internal, and External Fragmentation, Compaction, Paging: Principle of Operation, Page Allocation – Hardware Support for Paging, Protection and Sharing, Disadvantages of Paging. Virtual Memory: Basics of Virtual Memory – Hardware and Control Structures – Locality of Reference, Page Fault, Working Set, Dirty Page/ Dirty Bit – Demand Paging, Page Replacement Algorithms: Optimal, First In First Out (FIFO), and Least Recently Used (LRU).

Unit-VI File System and I/O Management — 06 Hrs.

File Management: Concept of File, Access Methods, File Types, File Operation, Directory Structure, File System Structure, Allocation Methods (Contiguous, Linked, Indexed), Free-Space Management (Bit Vector, Linked List, Grouping), Directory Implementation (Linear List, Hash Table), Efficiency and Performance. Input /Output Management: I/O Management and Disk Scheduling, I/O Devices, I/O Buffering, Disk Scheduling Algorithm: FCFS, SSTF, SCAN, CSCAN, LOOK, C-LOOK.

Reference Books

1. Dhananjay M. Dhamdhere, "Operating Systems-A Concept-Based Approach", 3rd edition, TMH, 2012.

2. Achyut Godbole and Atul Kahate, "Operating Systems", 3rd Edition, McGraw Hill Education, 2017.

3. Maurice J. Bach, "Design of UNIX Operating System", 2nd Edition, PHI, 2004.

Text Books

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts", 9th Edition, John Wiley & Sons, Inc., 2016.

2. William Stallings, "Operating System: Internals and Design Principles", 8th Edition, Prentice Hall, 2014.

3. Andrew Tannenbaum, "Operating System Design and Implementation", 3rd Edition, Pearson, 2015.

Software Engineering (CMN401)

Teaching Scheme	Examination Scheme
Lectures : 03 Hrs./week	Theory : ISE : 40 Marks
Credits : 03	ESE : 60 Marks
	Total Marks : 100 Marks

Course Prerequisites:

- Basic Programming Skills
- Innovative Thinking.
- Enthusiasm to learn Management concepts

Course Objectives:

- To provide the idea of decomposing the given problem into Analysis, Design, Implementation, Testing and Maintenance phases.
- To provide an idea of using various process models in the software industry according to given circumstances.
- To gain the knowledge of how Analysis, Design, Implementation, Testing and Maintenance processes are conducted in a software project.

Course Outcomes:

After successful completion of the course, students will able to-

CO1: Understand the phases of Software development life cycle and Process models.

CO2: Understand the various kinds of software requirements.

CO3: Explain Requirement Engineering process and change management.

CO4: Understand the design concepts, design models.

CO5: Use the various testing techniques of a software system.

Course Contents

Unit-I Introduction — 08 Hrs.

Introduction to Software Engineering and Process Models: Introduction, Changing nature of software, Software myths.

Process Models: The Waterfall model, Incremental process models, Evolutionary process models, The Unified process, Agile process models.

Unit-II Software Requirements — 08 Hrs.

Software Requirements: Functional and Non-functional requirements, User requirements, System requirements, The Software requirements document.

Unit-III Requirement Engineering Process — 08 Hrs.

Requirement Engineering Process: Feasibility studies, Requirements elicitation and Analysis, Requirement validation, Requirement Management.

Unit-IV Design — 08 Hrs.

Design: Design process and Design quality, Design concepts-Abstraction, Information Hiding, Functional Independence, Refactoring, Modularity, Refinement, Design Classes, Design Model.

Unit-V Testing — 08 Hrs.

Testing:Testing Strategies-A Strategic approach to Software testing, Test strategies for Conventional software, White Box Testing, Black Box Testing, Validation Testing, System Testing.

Reference Books

1. Pankaj Jalote, "Software Engineering, A Precise Approach", International Edition,Wiley India, 2010.
2. Waman S Jawadekar, "Software Engineering: A Primer", 1st Edition, Tata McGraw-Hill, 2008.
3. Rajib Mall,"Fundamentals of Software Engineering", 3rd Edition, PHI, 2005.

Text Books

1. Roger S.Pressman, "Software Engineering, A Practitioner's Approach", 7th Edition, Mc Graw Hill, International Edition.
2. Sommerville, "Software Engineering", 7th Edition, Pearson education.