Khandesh Shikshan Mandal Sanchalit,

Pratap College, Amalner

(Autonomous)

Dist.Jalgaon Maharashtra

DST-FIST assisted College | UGC honored "A College with Potential For Excellence"

NAAC re-arcredated 'A+' Grade with CGPA 3.52 | RUSA Funded

'A+' Grade NAAC
Reaccredited (CGPA 3.52)
DST-FIST Assisted College
UGC Honored "A College with Potential for Excellence"

Syllabus for

T. Y. B. Sc. Computer Science

Under New Education Policy-2020

(With effect from June 2025)

Semester-wise Code structure for

B. Sc. (Honors/Research)

Computer Science Program

As per NEP 2020,

w. e. f. – June 2025

Paper Name	Credits	No. of Lect in (Hr. per	Teaching Learning Method	Total Marks		External (UA)		Internal (CA)	
				Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks
SEM V									
1. Major Core Subject Group									
CS-MJ-301 Database Management System	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-MJ-302 Software Engineering	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-MJ-303 Python Programming-I	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-MJP-304 Lab on RDBMS	**2**	**4**	**Practical**	50	20	30	12	20	8
CS-MJP-305 Lab on Python Programming-I	**2**	**4**	**Practical**	50	20	30	12	20	8
2. Vocational/Skill Enhancement course Group									
CS-VSCP-306 Lab on Digital Graphics and Media Design	**2**	**4**	**Practical**	50	20	30	12	20	8
3. Minor Subject Group *(Select any One)*									
CS-MN-311 Python Programming	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-MNP-312 Lab on Python Programming	**2**	**4**	**Practical**	50	20	30	12	20	8
4. Discipline Elective course Group									
CS-EC-321 Java Programming-I	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-EC-322 Lab on Java Programming-I	**2**	**2**	**Practical**	50	20	30	12	20	8
5. Field Project									
CS-FP-341	**4**	**8**	**Practical**	100	40	60	24	40	16

SEM VI									
1. Major Core Subject Group									
CS-MJ-351 Operating System	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-MJ-352 Computer Networks	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-IKS-353 Cryptography and Indian Knowledge Systems	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-MJP-354 Lab on Python Programming-II	**2**	**4**	**Practical**	50	20	30	12	20	8
CS-MJP-355 Lab on Operating System	**2**	**4**	**Practical**	50	20	30	12	20	8
2. Vocational/Skill Enhancement course Group									
CS-VSCP-356 Practical based on CASE Tool and Software Testing	**2**	**4**	**Practical**	50	20	30	12	20	8
3. Minor Subject Group *(Select any One)*									
CS-MN-361 Java Programming	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-MNP-362 Lab on Java Programming	**2**	**4**	**Practical**	50	20	30	12	20	8
4. Discipline Elective course Group									
CS-EC-371 Java Programming-II	**2**	**2**	**Theory**	50	20	30	12	20	8
CS-ECP-372 Lab on Java Programming-II	**2**	**2**	**Practical**	50	20	30	12	20	8
5.Field Project									
CS-OJT-391	4	**8**	**Practical**	100	40	60	24	40	16

SEM-V

CS MJ 301: Database Management Systems

Course Objectives

- To teach fundamental concepts of RDBMS.
- To teach database management operations.
- Be familiar with the basic issues of transaction processing and concurrency control.
- To teach data security and its importance.

Course Outcomes

On completion of the course, student will be able to–

- Design E-R Model for given requirements and convert the same into database tables.
- Use database techniques such as SQL & PL/SQL.
- Explain transaction Management in relational database System.
- Use advanced database Programming concepts

Unit 1 INTRODUCTION TO DBMS

Introduction to DBMS, Tables, Database etc.
Database Operations - 1.Creating a Database 2.Dropping the Database Table Operations – 1. Create 2. Alter 3. Drop
Introduction to Open Source software,
Installation of open source software on Windows and Linux.

Unit 2 SQL – STATEMENTS, OPERATORS, FUNCTIONS

Statements - SELECT, INSERT, UPDATE, DELETE
Null value and Default value
Operators - Arithmetic, Logical, Comparison, Bitwise, Relational
Functions - Aggregate functions, Date and Time functions, String functions
Clauses: - where, order by, AND, OR, Between, Like, CASE, Distinct, Group by Having

Unit 3 VIEW, JOIN and DATA CONSTRANTS in SQL

Constraints - Data Integrity, Entity Integrity
Keys - PRIMARY KEY, UNIQUE, FOREIGN KEY, CHECK, Not Null
Views - Create, Alter, Drop
Join - Joins, Cross Join, Inner Join, Outer Join, Self-Join
Subquries -Subqueries as Constants, Subqueries as Correlated Values, Subqueries as Lists of Values, NOT IN and Subqueries with NULL Values, Subqueries Returning Multiple Columns
Statement - MERGE Statement
Set operations-UNION, EXCEPT, and INTERSECT Clauses -ANY, ALL, and EXISTS Clauses

Unit 4 Transaction Commands, Indexes & Sequences

Transaction commands-Commit,
Rollback Indexing -Creating an
Index, Unique Indexes
Sequences- Creating Sequence, using nextval(), currval() and setval()

REFERENCE BOOKS:

1. Bruce Momjian , PostgreSQL Introduction and Concepts, Addison.Wesley, ISBN 0- 201-70331-9
2. NEIL MATTHEW AND RICHARD STONES **,** Beginning Databases with PostgreSQL, From Novice to Professional, Second Edition, ISBN (pbk): 1-59059- 478-9
3. Sudhakar Bhoite," RDBMS With Oracle Developer 2000", ISBN: 978-8189065164
4. Himanshu Dabir, Dipali Meher "Advanced RDBMS Using Oracle", ISBN:978-9350161500, Vision Publications; Second edition.

CS-MJ-302: Software Engineering

Course Objectives:

This paper helps to understand

- What is software and the process in development of software
- It gives detailed knowledge about various models and requirements needed in developing software.
- It also elaborates the concepts of designing, testing & quality about software.

Course Outcomes:

After completion of the course:

- Students are able to perform the E-R Diagram, DFD, Data dictionary, Decision tree about software.
- They can also design the software in learned language using the course content.
- Get the knowledge of types of testing & how testing is performed in industry.

1. **Introduction to Software Engineering**
   Software and Software Engineering Evolution of Software
   Software Characteristics Software Applications Software Myths
   Software Process
   Software Development Life Cycle (SDLC)
2. **Software Development Model**
   Waterfall Model Prototyping Model
   Incremental Development Model RAD model
   Spiral Model
3. **Requirement Analysis and Specification**
   Requirements Engineering Fact finding Techniques
   Introduction to Types of Requirement Modeling
   Data Modeling Concepts- Data Objects, Data Attributes & Relationship.
4. **Design Engineering**
   Characteristics of good Software Design
   Design Concepts- Architecture, Modularity, Information Hiding Cohesion & Coupling
   Decision Table & Decision Tree Data flow Diagram
   Data Dictionary
5. **Software Coding & Testing**
   Coding standards & Guidelines
   What is testing? Testing Activities Black box testing White box testing
   Introduction to Debugging Approaches – Brute force Method, Backtracking, Case Elimination Method, Programming Slicing
6. **Software Quality**
   What is Quality?
   Software Quality - Garvin's quality dimensions, Mc Calls quality factors, ISO 9125 quality factors
   Elements of Software Quality Assurance ISO 9000 & Certification

References –

a. Roger S. Pressman, "Software Engineering a Practitioners Approach", ISBN 13: 9780071267823, 7 th edition, McGraw Hill International Edition.
b. Rajib Mall, "Fundamental of Software Engineering", ISBN- 978-81-203- 3819-7 3 RD Edition,, PHI Learning Private Limited.
c. Gaurav Gupta and Deepika Gupta, "Software Engineering", ISBN:978-3843310130, Lambert Publishing pvt. Ltd
d. Udit Agrawal, "Software Engineering", ISBN: 978-9350142363

CS-MJ-303: Python Programming-I

Course Objectives:

- The course is designed to provide Basic knowledge of Python.
- Python programming is intended for software engineers, system analysts, program managers and user support personnel who wish to learn the Python programming language.
- To learn how to design and program Python applications.
- To develop problem solving skills and their implementation through Python.
- Master the fundamentals of writing Python scripts

Course Outcome:

- Explain basic principles of Python programming language
- Construct and apply various filters for a specific task.
- Apply the best features of mathematics, engineering and natural sciences to program real life problems.

Unit – 1 Introduction to Python Programming

Introduction to Python History of Python Version of Python
Need, Features of Python Applications of Python
Installing Python on Linux and Windows Installing Python IDE

Unit – 2 Basics of Python Programming

Python Identifiers, Variables and Keywords Putting Comments Expressions and Statements
Standard Data Types – Basic, None, Boolean, Numbers. Type Conversion
Function Operators in Python Operator Precedence
Accepting Input and Displaying Output
Flow Control Statements Conditional Statements Looping Statements break, continue, pass Statements

Unit – 3 Python Strings

Introduction to String, String Literals, Assign String to a Variable Multiline Strings
Operations on Strings, Index Operator: Working with the Characters of a String, String Length, the Slice Operator, String Comparison,
Concepts of Python Lists: Creating, Initializing and Accessing elements in lists,Traversing, Updating and deleting elements from Lists.
List Operations: Concatenation, List Indexing, Slices Built- in List functions and methods, Aliasing, Cloning Lists

Unit – 4 Python Tuples and Dictionary

Introduction to Tuples Creating Tuples. Deleting Tuples, Accessing elements in a Tuple, Tuples Operations: Concatenation, Repetition, Membership, and Iteration. Built- in Tuples functions and methods
Introduction to Dictionary
Dictionaries: Concept of key-value pair.
Creating, Initializing and Accessing elements in a Dictionary. Traversing, Updating and Deleting elements in a Dictionary Built- in Dictionary functions and methods

Unit – 5 Python Functions and Modules

Introduction to Functions Defining a Function (def) Calling a Function, Function Arguments - Required arguments, Keyword arguments, Default arguments, Variable-length arguments Scope of Variables, Void functions and function returning values Recursion, Advance Function Topics: Anonymous Function Lambda, Mapping Functions, and Functional Programming Tools: filter and reduce

Introduction to Modules Creating Modules and Packages Importing Modules Using the dir() Function Built-in Modules.

References:

1. John V Guttag (2013), Introduction to Computation and Programming Using Python, Prentice Hall of India, 2013, ISBN: 9780262525008
2. Peter C. Norton, Alex Samuel and others, ―Beginning Python‖, Wrox Publication,2005 ISBN 10: 0764596543 ISBN 13: 9780764596544
3. R. Nageswara Rao(2016), Core Python Programming, Dreamtech Press, 2016, ISBN-13: 9789351199427
4. Wesley J. Chun(2006), Core Python Programming - Second Edition, Prentice Hall, ISBN- 13: 978-0132269933, ISBN-10: 0132269937
5. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser(2013), Data Structures and Algorithms in Pyhon'', Wiley, 2013, ISBN : 978-1-118-54958-2, ISBN : 978-1-118- 29027-9 (HardCover)
6. Luke Sneeringer(2015), Professional Python, Wiley Inc.,2015, ISBN: 1119070856

CS-MJP-304: Lab on RDBMS

1. To create one or more tables with following constraints, in addition to the two constraints (PK & FK)
   a. Check constraint
   b. Unique constraint
   c. Not null constraint
2. To drop a table, alter schema of a table, insert / update / delete records using tables created in previous Assignments. (use simple forms of insert / update / delete statements)
3. To query the tables using simple form of select statement Select <field-list> from table [where <condition> order by <field list>] Select <field-list, aggregate functions > from table [where <condition> group by <> having <> order by <>]
4. To query table, using set operations (union, intersect)
5. To query tables using nested queries.
6. To create views.
7. To create Stored Procedure
8.
   - A Simple Stored Procedure
   - A Stored Procedure with IN, OUT and IN/OUT parameter
9. Stored Function
   - A Simple Stored Function
   - A Stored Function that returns

CS-MJP-305: Lab on Python Programming-I

1. Installing python and setting up environment. Simple statements like printing the names (“Hello World”), numbers, mathematical calculations, etc.
2. Write a program to find all prime numbers within a given range.
3. Write a program to print "n" terms of Fibonacci Series using Iteration
4. Write a program to demonstrate the use of slicing in string.
5. Programs related to string manipulation
6. Write a Programs related to functions & modules
7. Write a program that demonstrate concept of functional programming.
8. Write a program to demonstrate the use of list & related functions
9. Write a program to demonstrate the use of Dictionary& related functions
10. Write a program to demonstrate the use of tuple.

CS VSCP 306: Lab on Digital Graphic's & Media Design

Category A: Programming-Based Computer Graphics Practical's

1. **Draw Basic Shapes Using Graphics Library**
   Objective: Use a graphics library (e.g., graphics.h in C++) to draw lines, rectangles, circles, and polygons.
2. **Implement Digital Line Drawing Algorithms**
   Objective: Implement **DDA** and **Bresenham's Line Drawing Algorithm** to plot lines pixel-by-pixel.
3. **Basic Animation Using Graphics Library**
   Objective: Create a simple animation (e.g., moving car, bouncing ball) using frame-by-frame rendering.

Category B: Pixel-Based Practical's

4. **Pixel Color Manipulation of an Image**
   Program to implement Indian Flag.
5. **Create Pixel Art Using Grid-Based Drawing**
   Objective: Design a simple image (like an emoji or character) using pixel grid logic in code.

Category C: Design-Based Practical's Using Canva

6. **Design a Digital Poster Using Canva**
   Objective: Create a visually appealing poster for an event, using layers, typography, and graphics.
7. **Design a Social Media Template or Resume in Canva**
   Objective: Build a reusable template for Instagram/Facebook post or a modern tech resume using Canva's tools.

CS-MN-311: Python Programming

Course Objectives:

- The course is designed to provide Basic knowledge of Python.
- program managers and user support personnel who wish to learn the Python
- To learn how to design and program Python applications.
- To develop problem solving skills and their implementation through Python.
- Master the fundamentals of writing Python scripts

Course Outcome:

- Explain basic principles of Python programming language
- Construct and apply various filters for a specific task.
- Apply the best features of mathematics, engineering and natural sciences to program real life problems.

Unit – 1 Introduction to Python Programming

Introduction to python History of Python Version of Python, Need, Features of Python
Applications of Python, Installing Python on Linux and Windows Installing Python IDE

Unit – 2 Basics of Python Programming

Python Identifiers, Variables and Keywords Putting Comments, Expressions and Statements
Standard Data Types – Basic, None, Boolean, Numbers. Type Conversion Function Operators in Python Operator Precedence, Accepting Input and Displaying Output
Flow Control Statements Conditional Statements Looping Statements break, continue, pass Statements

Unit – 3 Python Strings

Introduction to String, String Literals, Assign String to a Variable Multiline Strings Operations on Strings, Index Operator: Working with the Characters of a String, String Length, the Slice Operator, String Comparison, Concepts of Python Lists: Creating, Initializing and Accessing elements in lists,Traversing, Updating and deleting elements from Lists, List Operations: Concatenation, List Indexing, Slices Built- in List functions and methods, Aliasing, Cloning Lists

Unit – 4 Python Functions and Modules

Introduction to Functions Defining a Function (def) Calling a Function, Function Arguments - Required arguments, Keyword arguments, Default arguments, Variable-length arguments Scope of Variables Void functions and function returning values Recursion
Introduction to Modules Creating Modules and Packages Importing Modules Using the dir() Function Built-in Modules

Unit – 5 Object Oriented Concepts in Python

Overview of OOP Terminology Creating Classes, Creating Instance Objects Accessing Attributes Built-In Class Attributes, Garbage Collection: Constructor Overloading Methods and Operator, Inheritance - Implementing a subclass, Overriding Methods

References:

1. John V Guttag (2013), Introduction to Computation and Programming Using Python, Prentice Hall of India, 2013, ISBN: 9780262525008
2. Peter C. Norton, Alex Samuel and others, ―Beginning Python‖, Wrox Publication,2005 ISBN 10: 0764596543 ISBN 13: 9780764596544
3. R. NageswaraRao(2016), Core Python Programming, Dreamtech Press, 2016, ISBN-13: 9789351199427
4. Wesley J. Chun(2006), Core Python Programming - Second Edition, Prentice Hall, ISBN- 13: 978-0132269933, ISBN-10: 0132269937
5. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser(2013), Data Structures and Algorithms in Pyhon", Wiley, 2013, ISBN : 978-1-118-54958-2, ISBN : 978-1-118- 29027-9(HardCover)
6. Kenneth A. Lambert(2011), Fundamentals of Python – First Programs, CENGAGE

CS-EC-321 Java Programming-I

Course Objectives:

- To learn Object Oriented Design with JAVA
- Ability to write computer program to solve specific program
- To handle abnormal termination of a program using exception handling

Course Outcomes:

- Get knowledge of JDK environment
- Explore polymorphism using method overloading and method overriding
- Understand the different aspects of hierarchy of classes and their extensibility
- Understands the concept of streams and files
- Write programs for handling run time errors using exceptions

Unit-1 Introduction to JAVA

History of Java
Comparison of Java and C++
Features - Simple, Object Oriented Distributed, Robust, Secure, Architecture neutral, Portable, Interpreted, High Performance, Multithreading, dynamic.
Java and Internet
JDK Environment (Java, Javac, Applet Viewer, Javadoc)

Unit-2 Basics of JAVA

Variables, Data Types, Casting, Operators Compiling and running java program, Command line rguments. Data members, methods, Accepting input from console (Using BufferedReaderclass, Scanner) Control Statements, Looping Statements, Arrays

Unit-3 Objects and Classes

Introduction – Classes and Objects
Types of Constructors Overloading Packages
Access modifier Inner classes
String functions - Concatenation, Substring, String editing, Testing for Equality, Character extraction functions – CharAt, getChars, getByte

Unit-4 Object Oriented Programming Principles

Inheritance, Types of Inheritance
Polymorphism- Overloading & Overriding
Abstraction-Interface & Abstract Class
Encapsulation

Unit-5 Exception Handling

Dealing with errors - Types of exceptions
Exception Hierarchy
Exception Handling Mechanism Catching Exceptions.
Creating user defined exception

Unit-6 Streams and Files

String class and String Buffer Class Using the File class
Stream classes-Byte Stream classes, Character Stream Classes Creation of files
Reading/Writing characters and bytes handling primitive data types Random Access files

References:

2. Cay's Horstmann and Gary Cornell, " Core Java Volume -1 Fundamentals", ISBN: 81-7808-277-2
3. E. Balaguruswamy ,"Programming with Java – A primer", ISBN: 978-0-07-061713-1
4. Herbert Schildt, "The complete reference JAVA-2", ISBN:978-0-07-049543-2, Fifth Edition,(TMH)
5. Java 6 Programming Black Book.
6. Horstmann, C. S/ Cornell, "Core Java 2: Volume II – Advanced Features", 7th ed Pearson.

CS-EC-322: Lab on Java Programming-I

1. Write a simple program in Java to print first fifty prime number.
2. Write a program in Java to print factorial of given number using recursion
3. Write a program in Java to create student information using array
4. Write a program in Java to implement user defined package.
5. Write a program in Java to implement default & parameterized constructor.
6. Write a program in Java to demonstrate various operations on string functions.
7. Write a program in Java to demonstrate class.
8. Write a program in Java to implement inheritance.
9. Write a program to demonstrate User defined package in java
10. Write a program to demonstrate Interface and Abstract Class

CS-MJ-351 Operating System

Objectives:

- To understand Operating system concepts and services.
- To understand the concept of a CPU scheduling, memory management, Disk Drum Scheduling and deadlock.

Outcomes:

- Students should familiar with Operating System Services.
- Understand CPU scheduling algorithms, memory Management Techniques, Disk Drum Scheduling algorithms, Deadlock preventions and avoidance.
- Introduction to android operating systems – its architecture, applications and uses.

Unit 1.Introduction

What is an Operating system? Types of Operating System Services of Operating System Functions of operating system.

Unit 2.CPU scheduling

Multiprogramming Concepts, Basic Concept of CPU scheduling: CPU-I/O burst cycle, CPU scheduler, Scheduling Algorithms: FCFS, SJF, Priority scheduling, Round-robin scheduling Multilevel queues, multilevel feedback queue

Unit 3.Memory Management

Logical versus Physical Address space Swapping, Multiple partition allocation MFT , MVT Paging, Segmentation, Virtual Memory Management – Background, Demand paging

Unit 4. Disk and Drum Scheduling

First Come first serve, scheduling Shortest Seek Time First Scheduling SCAN Scheduling C-SCAN Scheduling

Unit 5 Deadlocks

Concept of Deadlock Characterization Deadlock Prevention Deadlock Avoidance Deadlock Detection

Reference books:

1. Peterson Silberschatz, “Operating system concepts”, ISBN: 0-201-35251-6, Addison Wesley, 1ST Edition
2. Andrew S. Tanenbaum, “Modem operating system”, ISBN: 81-203-0974-X, P .H.I. New Delhi 3.
3. Achyut S. Godbole, “Operating Systems” ISBN: 9780070702035, McGraw Hill Education, 2010, Third Edition

CS-MJ-352 Computer Networks

Course Objectives:

This paper helps to understand:

- How network works? & types of networks & its applications.
- It helps to understand the various models.
- It helps to understand various layers & their functionality.
- It get the idea of how cryptography works.

Course Outcomes:

After completion of the course:

- Students understand the information exchange done across the network with thehelp of OSI & TCP/IP models.
- Student understands how errors are captured & handled in network.
- Student understands various attack & its prevention techniques.

Unit-1 Introduction to Computer Network and Network Model

What is Computer Network? Application of Computer Networks Transmission Mode, Network Structure Network Topologies
ISO OSI Reference Models, TCP / IP Reference Model & their Comparison.

Unit-2 Physical Layer

Guided Media: Twisted Pair Coaxial Cable Fiber Optics
Satellite Communication Microwave Communication
Radio Transmission Microwave Transmission Infrared & Millimeter Waves Light wave

Unit 3 The Data link Layer

Services Provided to Network Layer Framing, Error Control , Flow Control
Error Detection – Redundancy, Parity Check, Checksum & CRC Error Correction – Hamming Code.

Unit 4 The Network Layer

Logical Addressing, IP v4 Addresses - Address Space - Classful Addressing - Classless Addressing Routing Algorithm, Shortest Path Multicast Routing, Congestion Control, Introduction to Congestion Control

Unit-5 Transport Layer

Process to Process Delivery Client, Server Paradigm, Multiplexing and DE multiplexing Connectionless v/s Connection Oriented Services Reliable v/s Unreliable Transmission
UDP and TCP UDP – Operations and uses TCP – Services and features

Reference Books: -

1. Andrew S.Tanenbaum , "Computer Networks " ISBN: 978-0130661029, Prentice Hall, Fourth Edition .
2. Behrouz A. Forouzan , "Data Communication & Networking", ISBN: 978-0071232418 , McGraw Hill Higher Education , Third Edition 3.
3. U.D. Black , "Data Communication & Distributed Networks", ISBN: 9780835913416, Published by
   Prentice-Hall, Englewood Cliffs, N.J., 1987 , Second Edition ,
4. Atul Kahate , " Cryptography and Network Security " Edition 3, McGraw Hill.

CS-IKS-353 Cryptography and Indian Knowledge Systems

Unit 1: Introduction to Cryptography in Indian Context

- Basics of cryptography: encryption, decryption, keys, ciphers.
- Need for secrecy in governance, trade, and warfare in ancient India.
- References from **Arthashastra (Kautilya's Chanakya Niti)** – use of secret writing, spies, and coded communication.
- Examples of traditional secret codes (Mudra Lipi, coded scripts in manuscripts).

Unit 2: Ancient Indian Methods of Cryptography

- Katapayadi System (encoding numbers into syllables).
- Mantra-based encoding – phonetic patterns for hidden meanings.
- Bhaskaracharya's works – hidden numerical patterns in poetry.
- Palm-leaf manuscripts & methods to protect sacred knowledge.

Unit 3: Mapping Ancient Systems to Modern Cryptography

- Steganography in Indian art, poetry, and literature.
- Cryptographic parallels: substitution & transposition ciphers in Indian texts.
- Use of Sanskrit shlokas as hash-like patterns (fixed structure, tamper-proof).
- Comparison of **ancient Indian ciphers vs. Caesar cipher / Vigenère cipher**.

Unit 4: Modern Applications

Symmetric & Asymmetric cryptography overview.

- Indian knowledge-inspired algorithms (example: rule-based encoding like Panini's grammar).
- Blockchain security & parallels with **Sabha consensus in Vedic tradition**.

CS-MJP-354: Lab on Python Programming-II

1. Write a program to demonstrate Exception Handling mechanism
2. Write a program to demonstrate Regular expression in python.
3. Write a program to demonstrate the working of classes and objects.
4. Write a program to demonstrate the working of Inheritance and Overloading Methods and Operator.
5. Write a program to demonstrate read & write file.
6. Write a program to demonstrate to Renaming, Moving, Copying, and Removing Files,
7. Write a program to demonstrate to learn GUI programming using Tkinter.
8. Write a program to create a database application for insert, update and delete in a table using MySQL.

CS-MJP-355: Lab on Operating System

1. Write a C program to simulate the following non-preemptive CPU Scheduling algorithms to find turnaround time and waiting time.
    a) FCFS
    b) SJF

2. Write a C program to simulate the Round Robin (pre-emptive) CPU Scheduling algorithms to find turnaround time and waiting time.

3. Write a C program to simulate the Following memory management Techniques.
    a) MVT b) MFT

4. Write a C program to simulate paging technique of memory management.

5. Write a C program to simulate disk scheduling algorithms
    a) FCFS b) SCAN

CS-MN-361: Java Programming

Course Objectives:

- To learn Object Oriented Design with JAVA
- Ability to write computer program to solve specific program
- To handle abnormal termination of a program using exception handling

Course Outcomes:

- Get knowledge of JDK environment
- Explore polymorphism using method overloading and method overriding
- Understand the different aspects of hierarchy of classes and their extensibility
- Understands the concept of streams and files
- Write programs for handling run time errors using exceptions

Unit-1 Introduction to JAVA

History of Java, Comparison of Java and C++
Features - Simple, Object Oriented Distributed, Robust, Secure, Architecture neutral, Portable, Interpreted, High Performance, Multithreading, dynamic, Java and Internet, JDK Environment (Java, Javac, Applet Viewer, Javadoc)

Unit-2 Basics of JAVA

Variables, Data Types, Casting, Operators Compiling and running java program, Command line arguments.
Data members, methods, Accepting input from console (Using BufferedReaderclass, Scanner)
Control Statements, Looping Statements, Arrays

Unit-3 Objects and Classes

Introduction – Classes and Objects, Types of Constructors Overloading Packages, Access modifier Inner classes, String functions - Concatenation, Substring, String editing, Testing for Equality, Character extraction functions – CharAt, getChars, getByte

Unit-4 Object Oriented Programming Principles

Inheritance, Types of Inheritance, Polymorphism- Overloading & Overriding, Abstraction- Interface & Abstract Class, Encapsulation

Unit-5 GRAPHICS Programming

Introduction- frames, frame layouts
Displaying information in a frame, Graphics objects and paint component method Text and Fonts, Colors, Drawing Shapes, Filling Shapes Paint mode and Images.

References:

1. Cay's Horstmann and Gary Cornell ,"Core Java Volume 2", ISBN: 978-0-13- 708160-8, 9TH edition, published by Prentice Hall
2. E. Balaguruswamy ,"Programming with Java – A primer", ISBN:978-0-07-061713-1
3. Herbert Schildt, "The complete reference JAVA-2", ISBN: 978-0-07-049543-2, FifthEdition,(TMH)
4. Java Programming Black Book.

Buyya, Selvi, Chu, , "Object Oriented Programming with Java", ISBN: 978-0070678835, TataMcGraw Hill Education2010

CS-MNP-362: Lab on Java Programming

1. Write a simple program in Java to print first fifty prime number.
2. Write a program in Java to print factorial of given number using recursion
3. Write a program in Java to create student information using array
4. Write a program in Java to implement user defined package.
5. Write a program in Java to implement default & parameterized constructor.
6. Write a program in Java to demonstrate various operations on string functions.
7. Write a program in Java to demonstrate class.
8. Write a program in Java to implement inheritance.
9. Write a program in Java to display messages in various fonts in a frame

CS-EC-371 Java Programming-II

Course Objectives:

- To design User Interface using Swing and AWT
- Learn the advanced concept of java
- To aware about the applet programming

Course Outcomes:

- ✓ Program using graphical user interface with Swing classes
- ✓ Handle different kinds of events generated while handling GUI components
- ✓ Create programs using menus and dialog boxes
- ✓ Program to create applets
- ✓ Understand advanced java concepts like JDBC, Java Beans

Unit-1 GRAPHICS Programming

Introduction- frames, frame layouts
Displaying information in a frame, Graphics objects and paint component method Text and Fonts, Colors, Drawing Shapes, Filling Shapes Paint mode and Images.

Unit-2 Event Handling

Event Handling Mechanism
Concept: AWT, Swing, Difference between AWT and Swing. The AWT event hierarchy
Event handling summary- event sources and listener, adapter classes. Low level events - Focus, window, keyboard, mouse events.
Multicasting

Unit-3 User Interface Components Using SWING

Introduction to layout management - Panels, Border Layout, Grid Layout, Text Input- Text Field, Text Area, Password field
Labels and Buttons, Making choices - Check boxes, Radio buttons, List, Combo boxes

Unit-4 Menu and Dialog Box

Menus – Building menus Menu events, Popup menu, Keyboard mnemonics and Accelerators, enabling and disabling menus Dialog boxes - opening dialogs using inbuilt dialog box

Unit-5 Introduction to Advanced JAVA

Collections, Interfaces- List, Set, Queue.
Classes- Array List, Vector
Database connectivity -JDBC

References:

1. Cay's Horstmann and Gary Cornell ,"Core Java Volume 2", ISBN: 978-0-13- 708160-8, 9THedition, published by Prentice Hall
2. E. Balaguruswamy ,"Programming with Java – A primer", ISBN:978-0-07-061713-1
3. Herbert Schildt, "The complete reference JAVA-2", ISBN: 978-0-07-049543-2, FifthEdition,(TMH)
4. Java Programming Black Book.
5. Buyya, Selvi, Chu, , "Object Oriented Programming with Java", ISBN: 978- 0070678835, TataMcGraw Hill Education2010

CS-ECP-372 Lab on Java Programming-II

1. Write a program in Java to display messages in various fonts in a frame
2. Write a program in Java to draw various geometric shapes like circle, line, rectangle etc.
3. Write a program in Java to demonstrate paint mode.
4. Write a program in Java to demonstrate window events.
5. Write a program in Java to demonstrate Mouse events.
6. Write a program in Java to demonstrate Keyboard events.(key pressed, key released)
7. Write a program in Java to demonstrate multicasting
8. Write a program in Java to demonstrate user interface component list boxes and combo box.
9. Write a program in Java to demonstrate user interface component radio button and check box.
10. Write a program in Java to demonstrate menus as interface component.
11. Write an Applet to display human face.
12. Write a program in Java to demonstrate Java Applet with parameter