AC: 02.06.2025 ITEM NO: 2.2

Deccan Education Society's

Kirti M. Doongursee College of Arts, Science and Commerce (AUTONOMOUS)

Affiliated to

UNIVERSITY OF MUMBAI

Syllabus for
Program: Bachelor of Science
Course: S.Y.B.SC.
Subject: Information Technology

Choice Based Credit System (CBCS)
with effect from
Academic Year 2024-2025

PROGRAM OUTCOMES

PO	Description
A student completing Bachelor's Degree in **Science** Program will be able to	
PO1	Disciplinary Knowledge: Demonstrate comprehensive knowledge of the disciplines that form a part of a graduate Programme. Execute strong theoretical and practical understanding generated from the specific graduate Programme in the area of work.
PO2	Critical Thinking and Problem solving: Exhibit the skills of analysis, inference, interpretation and problem-solving by observing the situation closely and design the solutions
PO3	Social competence: Display the understanding, behavioral skills needed for successful social adaptation, work in groups, exhibits thoughts and ideas effectively in writing and orally.
PO4	Research-related skills and Scientific temper: Develop the working knowledge and applications of instrumentation and laboratory techniques. Able to apply skills to design and conduct independent experiments, interpret, establish hypothesis and inquisitiveness towards research
PO5	Trans-disciplinary knowledge: Integrate different disciplines to uplift the domains of cognitive abilities and transcend beyond discipline-specific approaches to address a common problem.
PO6	Personal and professional competence: Performing dependently and collaboratively as a part of a team to meet defined objectives and carry out work across interdisciplinary fields. Execute interpersonal relationships, self-motivation and adaptability skills and commit to professional ethics.
PO7	Effective Citizenship and Ethics: Demonstrate empathetic social concern and equity centered national development, and ability to act with an informed awareness of moral and ethical issues and commit to professional ethics and responsibility.
PO8	Environment and Sustainability: Understand the impact of the scientific solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development.

Information Technology Department

S.Y.B.Sc. Syllabus (NEP 2020)

Academic year 2024-25

Semester	Course Code	Course Title	Vertical	Credit
III	24ITMJ311	Computer Graphics & Animation	Major	2
	24ITMJP31	Computer Graphics & Animation Practical	Major	2
	24ITMJ312	Software Engineering	Major	2
	24ITMJP32	Software Engineering Practical	Major	2
	24ITMR321	Fundamentals of Statistics	Minor	2
	24ITMRP31	R Programming Practical	Minor	2
	24ITOE331	Descriptive Statistics	OE	2
	24ITVC341	Advanced Database Management System	VSC	2
	24ITFP3	Field Project	FP	2
IV	24ITMJ411	Core Java	Major	2
	24ITMJP41	Core Java Practical	Major	2
	24ITMJ412	Embedded System	Major	2
	24ITMJP42	Embedded System Practical	Major	2
	24ITMR421	Cryptography and Network Security	Minor	2
	24ITMRP41	Cryptography Practical	Minor	2
	24ITOE431	Probability Distribution	OE	2
	24ITSE441	Android Mobile Programming	SEC	2
	24ITCEP4	Community Engagement Program	CEP	2

SEMESTER- III

Course Code	Major SEM III	Credits	Lectures/ Week
24ITMJ311	**Paper I- Computer Graphics & Animation**	**2**	**2**

Course Outcomes:

After successful completion of this course, students would be able to

CO1:- State the applications, areas and graphic pipeline, display and storage technologies.

CO2:- Summarize the principles and techniques of Computer Animation.

CO3:- Apply and compare the algorithms for drawing 2D images

CO4:- Analyze the clipping algorithms with the transformation on 2D images.

Unit	Topics	No of Lectures
I	**Introduction to Computer Graphics:** Overview of Computer Graphics, Computer Graphics Application and Software, Display Technologies, Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Random-Scan Display Processor **Scan conversion** – Digital Differential Analyzer (DDA) algorithm, Bresenhams' Line drawing algorithm. Bresenhams' method of Circle drawing, Midpoint Circle Algorithm, Midpoint Ellipse Algorithm, Clipping Lines algorithms– Cyrus-Beck, Cohen-Sutherland and Liang-Barsky.	**15**
II	**Two-Dimensional and Three-Dimensional Transformations:** Overview, Transformations, Viewing and Modelling, Projections- Perspective and Parallel Projection, Orthographic and Oblique Parallel Projection. **Determination of Visible Surfaces Overview and Computer Graphics pipeline** **Computer Animation:** Principles of Animation, Key framing, Deformations, Character Animation, Physics-Based Animation, Procedural Techniques.	**15**

References:

Sr. No. Title Author/s Publisher Edition Year

1. Computer Graphics -Principles and Practice, J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes, Pearson, 2nd

2. Steve Marschner & Peter Shirley Fundamentals of Computer Graphics, CRC press, 4th 2016

3. Computer Graphics Hearn, Baker Pearson, 2nd

4. Principles of Interactive Computer Graphics, William M. Newman and Robert F. Sproull, TMH, 2nd

Course Code	Major SEM – III	Credits	Lectures/Week
24ITMJP31	**Paper I - Computer Graphics Practical**	**2**	**4**

Course Outcomes:

After successful completion of this course, students will have the knowledge and skill to:

CO1-List different types of transformations in computer graphics programs.

CO2-Understand and implement different computer graphics algorithms.

CO3-Extract scene with different clipping methods and its transformation to graphics display device.

CO4-Compare and evaluate different graphical techniques.

Unit	Topics
1	**Solve the following:** a. Study and enlist the basic functions used for graphics in C++ language. Give an example for each of them. b. Draw a coordinate axis at the center of the screen.
2	**Solve the following:** a. Divide your screen into four regions, draw a circle, rectangle, ellipse and half ellipse in each region with appropriate messages. b. Draw a simple hut on the screen.
3	**Draw the following basic shapes in the center of the screen :** i. Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line
4	**Solve the following:** a. Develop the program for the DDA Line drawing algorithm. b. Develop the program for Bresenham's Line drawing algorithm.
5	**Solve the following:** a. Develop the program for the midpoint circle drawing algorithm.
6	**Solve the following:** a. Develop the program for the mid-point ellipse drawing algorithm.
7	**Solve the following:** a. Write a program to implement 2D scaling. b. Write a program to perform 2D translation.
8	**Solve the following:** a. Perform 2D Rotation on a given object. b. Program to create a house like figure and perform the following operations. i. Scaling about the origin followed by translation. ii. Scaling with reference to an arbitrary point. iii. Reflect about the line y = mx + c.

9	**Solve the following:** a. Write a program to implement Cohen-Sutherland clipping. b. Write a program to implement Liang - Barsky Line Clipping Algorithm
10	**Solve the following:** a. Perform smiling face animation using graphic functions. b. Draw the moving car on the screen.

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	SEM III-Major 2	Credits	Lectures/Week
24ITMJ312	**Paper II-Software Engineering**	**2**	**2**

Course Outcome:
After successful completion of this course, students would be able to
CO1: Remember the basic concepts, principles, and methodologies of software engineering.
CO2: Understanding the various software process models, such as Waterfall, Agile, and Spiral and identify the importance of software quality and testing strategies.
CO3: Apply project management principles to plan and schedule a software development project
CO4: Analyze and implement project scheduling concepts, risk management associated with various types of projects and Understand the software development process

Unit	Topics	No of Lectures
I	**Introduction:** What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc. **Software Requirements:** Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements. **Socio-Technical System: E**ssential characteristics of socio technical systems, Emergent properties, System Engineering, components of systems such as organizations, people and computers, Dealing Legacy systems. **Software Processes:** Process and Project, Component Software Processes. Software Development Process Models. • Waterfall Model. • Prototyping. • Iterative Development. • Rational Unified Process. • The RAD Model • Time boxing Model. **Agile / Rapid software development**: Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods. **Requirements Engineering Processes:** Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements Management. System Models: Models and its types, Context Models, Behavioral Models, Data Models, Object Models, Structured Methods. Resource Allocation - Introduction, Nature of Resources using the UML - Class diagram, Object diagram, Use case diagram, Sequence diagram, Collaboration diagram, State chart diagram, Activity diagram, Component diagram, Deployment diagram.	**15**

II	**Critical system:** Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems. **System Design:** System/Software Design, Architectural Design, Low-Level Design Coupling and Cohesion, Functional-Oriented Versus Object-Oriented Approach, Design Specifications, Verification for Design, Monitoring and Control for Design **Software Measurement and Metrics**: Process Metrics and Project Metrics, Software Measurement, Object Oriented Metrics, Software Project Estimation, Decomposition Techniques, LOC based, FP based and Use case based estimations, Empirical estimation Models **Software Project Management**: Estimation in Project Planning Process, Software Scope and Feasibility, Resource Estimation, Empirical Estimation Models – COCOMO II, Estimation for Agile Development, The Make/Buy Decision **Project Scheduling:** Basic Principles, Relationship Between People and Effort, Effort Distribution, Time-Line Charts	**15**

References:

- Software Engineering, A Practitioner's Approach, Roger S, Pressman, 2019
- Software Engineering: principles and Practices, Deepak Jain, OXFORD University Press, 2008

Additional References:

- Software Engineering, Ian Sommerville, Pearson Education, 2017
- Fundamentals of Software Engineering, Fourth Edition, Rajib Mall, PHI, 2018
- Software Engineering: Principles and Practices, Hans Van Vliet, John Wiley & Sons, 2010
- A Concise Introduction to Software Engineering, Pankaj Jalote, Springer

Course Code	SEM – III-Major-2 Practical	Credits	Lectures/Week
24ITMJP32	**Paper II Software Engineering Practical**	**2**	**4**

Course Outcomes:

After successful completion of this course, students would be able to

CO1- Define various approaches to verification and validation of software including testing, measurements and estimation of software products

CO2- Understand software engineering

CO3- Apply software engineering principles

CO4- Analyze the characteristics, advantages, and limitations of different software development models to determine their suitability for specific software projects

Unit	Topics
1	Study and implementation of class diagrams.
2	Study and implementation of Use Case Diagrams.
3	Study and implementation of Entity Relationship Diagrams.
4	Study and implementation of Sequence Diagrams.
5	Study and implementation of State Transition Diagrams.
6	Study and implementation of Data Flow Diagrams.
7	Study and implementation of Collaboration Diagrams.
8	Study and implementation of Activity Diagrams.
9	Study and implementation of Component Diagrams.
10	Prepare timeline chart/Gantt Chart/PERT Chart.

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	Minor SEM –III	Credits	Lectures/Week
24ITMR321	Paper III - Fundamentals of Statistics	2	2

Course Outcomes:

After successful completion of this course, students would be able to

CO1:-Recalling basic statistical terms and definitions

CO2:-Explain the concepts of central tendency and dispersion and describe their applications in different situations.

CO3:- Construct tables and use graphs and diagrams to visually represent the data.

CO4:- Analyze the nature of data to evaluate the relationships and interpret the implications of the measures.

Unit	Topics	No of Lectures
I	**Data types**: attribute, variable, discrete and continuous variable. **Data presentation** : frequency distribution, histogram ogive, curves. **Measures of Central tendency:** Mean, Median, mode for raw data, discrete,grouped frequency distribution. **Measures dispersion**: Variance, standard deviation, coefficient of variation for raw data, discrete and grouped frequency distribution, quartiles, quantiles Real life examples **Moments :** Raw moments, central moments, relation between raw and central moments **Measures of Skewness and Kurtosis**: based on moments, quartiles, relation between mean, median, mode for symmetric, asymmetric frequency curve.	15
II	**Correlation and Regression**: bivariate data, scatter plot, correlation, nonsense correlation, Karl pearson's coefficients of correlation, independence. **Linear regression**: fitting of linear regression using least square regression, coefficient of determination, properties of regression coefficients (only statement) **Probability**: Random experiment, sample space, events types and operations of events **Probability definition**: classical, axiomatic, Elementary Theorems of probability (without proof) $0 \leq P(A) \leq 1$, $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ $P(A') = 1 - P(A)$ $P(A) \leq P(B)$ if $A \subseteq B$ Conditional probability, 'Bayes' theorem, independence, Examples on Probability	15

Reference:

- Fundamental of Mathematical Statistics, S. C. Gupta and V. K. Kapoor, Sultan Chand and Sons.

Additional References:

- Probability and Statistics for Engineers and Sciences by J. L. Devore, CENGAGE Learning., 8th edition
- Statistical Inference, George Casella and Roger L. Berger

Course Code	Minor SEM III	Credits	Lectures/Week
24ITMRP31	Paper III R Programming Practical	2	4

After successful completion of this course, students would be able to
CO1:- Recalling the definition of matrix and its operation
CO2:-Understand the R environment by downloading and installing the packages,
CO3:- Apply correlation analysis to different realistic situations using R
CO4:- Analyze the skewness and kurtosis of data sets to evaluate their distribution and identify the underlying data patterns.

Unit	Topics
Practicals using R	
1	Execute the basic commands, array, list and frames
2	Execute the statistical functions: mean, median, mode, quartiles, range, inter quartile range, histogram
3	Import the data from Excel/.CSV file and find mean, median, mode, quartiles, range, inter quartile range, histogram.
4	Import the data from Excel/.CSV file and find standard deviation, variance and covariance.
5	Import the data from Excel/.CSV file and find skewness and kurtosis.
6	Create a table in it
7	Plot scatter diagram and correlation
8	Perform Linear Regression
9	Create a Matrix using R and Perform the operations addition, subtraction, multiplication, transpose, inverse
10	Perform Hypothesis Testing

References:

- **Crawley, M. J. (2006). Statistics - An introduction using R. John Wiley, London 32**
- **Statistics Using R, Narosa Publishing House, New Delhi. Eighth Ed. East - West Press. Statistics, Vol. 1, 6th revised edition, The World Press Pvt. Ltd., Calcutta.**

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	OPEN ELECTIVE SEM – III	Credits	Lectures /Week
24ITOE331	Descriptive Statistics	2	2

Course Outcomes:

After successful completion of this course, students would be able to

CO1: Remembering descriptive statistical concepts

CO2: Understanding descriptive statistics concepts

CO3: Apply correlation and regression concepts.

CO4: Analyse and infer using the concept of probability

Unit	Topics	No of Lectures
I	**Data types**: attribute, variable, discrete and continuous variable **Data presentation** : frequency distribution, histogram ogive, curves, stem and leaf display Data Aggregation **Measures of Central tendency:** Mean, Median, mode for raw data, discrete,grouped frequency distribution. **Measures dispersion**: Variance, standard deviation, coefficient of variation for raw data, discrete and grouped frequency distribution, quartiles, quantiles Real life examples **Moments :** raw moments, central moments, relation between raw and central moments **Measures of Skewness and Kurtosis**: based on moments, quartiles, relation between mean, median, mode for symmetric, asymmetric frequency curve.	15
II	**Correlation and Regression**: bivariate data, scatter plot, correlation, nonsense correlation, Karl pearson's coefficients of correlation, independence. **Linear regression**: fitting of linear regression using least square regression, coefficient of determination, properties of regression coefficients (only statement) **Probability**: Random experiment, sample space, events types and operations of events **Probability definition**: classical, axiomatic, Elementary Theorems of probability (without proof) $0 \leq P(A) \leq 1$, $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ $P(A') = 1 - P(A)$ $P(A) \leq P(B)$ if $A \subset B$ Conditional probability, 'Bayes' theorem, independence, Examples on Probability	15

References:

- Trivedi, K.S.(2001) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

- Ross, S.M. (2006): A First course in probability. 6 th Edⁿ Pearson
- Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): common statistical tests.Satyajeet Prakashan, Pune
- Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics,S. Chand and Sons, New Delhi
- Gupta, S.C. and Kapoor, V.K. (1999): Applied Statistics, S. Chand and Son's, New Delhi
- Montgomery, D.C. (2001): Planning and Analysis of Experiments, wiley.

Course Code	Vocational Skill Course	Credits	Lectures /Week
24ITVC341	**Paper I -Advanced Database Management System**	**2**	**4**

Course Outcomes:

After successful completion of this course, students would be able to

CO1- Recall fundamental concepts of database management systems, such as normalization and indexing.

CO2- Interpret the role of transaction management in a distributed environment.

CO3- Implement advanced SQL queries involving subqueries, joins, and nested queries.

CO4-Analyze the impact of concurrency control mechanisms on database performance.

Unit	Topics
1	Use of Joins in Database systems
2	Use of SubQueries in DBMS
3	Restricting and Sorting Data using clauses
4	Use of Embedded & Dynamic SQL
5	Use of With, Row_Number, Rank, Dense_Rank Functions
6	Apply Query Optimization
7	Practicals on Basics of PL/SQL
8	Write a UDF to use the RANK function and display desired results. Create 3 tables student1, student2, student3 with sName & sMarks columns having 5 distinct records each and write a User Defined Function to display the Student Names having top 5 marks
9	Write a UDF to solve a problem using date formats while accepting input from the user. Write a User Defined Function to display the difference, between your birthdate and current date, in "'NAME' was born 'YY' Years 'MM' Months 'DD' Days Ago" format by accepting 'NAME' from the user.
10	Write a PL/SQL block to use the loop and update the table as per conditions. Create a table student with 5 records in Name, Age, Adult columns and write a PL/SQL block to use the loop and update the Adult column with 'YES' and 'NO' depending on the Age column data.

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	FP	Credits	Hours
24ITFP3	**Field Project**	**2**	**60**

Course Outcomes:

After successful completion of this Field Project students would be able to

CO1: Recall the key principles and methodologies involved in implementing research projects.

CO2: Explain how theoretical knowledge can be applied to solve real-world problems through research projects.

CO3: Demonstrate the ability to implement a research project by applying theoretical concepts to practical scenarios.

CO4: Analyze project implementation processes to assess the effectiveness of teamwork, project management, and communication skills.

Field Project Guidelines

- **General Guidelines:**

1. Each department should ensure collaborations/Tie-ups (in terms of MoU/Lol) with relevant academic institutions/industries/organizations/NGO etc. as per project requirements.

2. All the communication with the academic institutions/industries/organizations etc. should be done through the department.

3. Internal faculty should be allotted to the students or group of students for the evaluation of the project.

4. Departments should maintain the relevant documents (such as attendance records, proposals, diary, MoUs/Lol etc) and correspondence regarding Field Project course.

- **Field Project (FP):**

Objectives:

a) To provide practical experience in implementing research projects.

b) To assess students' ability to apply theoretical knowledge in real-world situations.

c) To develop skills in project management, teamwork, and communication.

- 2 credits of Field Project comprises the ways of implementing actual field engagement which needs to be determined by respective departments.

	Credit	Student learning Hours	Course component
Sem III	2	60 Hrs.	Field Project

Note: Field Project 1 Credit = 30 Hours

- **Evaluation Consists of Two Parts:**

Evaluate each student for 50 marks per semester at department level -

- 20 marks for Continuous evaluation (CE)

- Review of project work to be undertaken.
- Progress report on project implementation.

-30 marks for End Semester Examination (ESE)

- Project Report
- Final presentation (PPT) of field project findings assessing project outcomes and reflection.

SEMESTER- IV

Course Code	SEM – IV- Major 1	Credits	Lectures/ Week
24ITMJ411	**Paper I-Core Java**	**2**	**2**

Course Outcomes:

After successful completion of this course, students would be able to

CO1- Identify classes, objects, members of a class and the relationships among them needed for a specific problem.

CO2- Understand the concept of OOP as well as the purpose and usage principles of inheritance, polymorphism, encapsulation and method overloading.

CO3- Use testing and debugging tools to automatically discover errors of Java programs as well as use versioning tools for collaborative programming/editing.

CO4- Relate object-oriented programming concepts in problem solving through Java.

Unit	Topics	No of Lectures
I	**Introduction:** Features of Java, Java Development Kit, Java Application Programming Interface, Java Virtual Machine, Object Oriented Programming (OOP) Principles, Control Statements. **Classes:** The Class Object and Its Attributes, Class Methods, Accessing A Method, Method Overloading, Instantiating Objects from A Class, Constructors, this keyword, static keyword, final keyword, Types of Classes, Scope Rules, Access Modifier, Garbage Collection. **Inheritance:** Basics, Multilevel Hierarchy, Method Overriding, super keyword, Abstract Classes, Abstract Methods.	**15**
II	**Packages & Interfaces:** Defining a Package, Package Access & its Import, What Is an Interface? How Is an Interface Different from An Abstract Class? Multiple Inheritance, Defining an Interface, Implementing Interfaces, Interfaces can be Extended, Default Interface Methods. **Exception Handling:** Fundamentals, Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause, Built-in Exceptions in java. **Multithreaded Programming:** Java Thread Model, Creating Thread, Creating Multiple Threads, Thread Priorities, Synchronization, InterThread Communication, Suspending, Resuming & Stopping Threads, Obtaining Thread's State.	**15**

References:

- Core Java 8 for Beginners Vaishali Shah, Sharnam Shah SPD 1st 2015
- Java: The Complete Reference Herbert Schildt McGraw Hill 9th 2014

Additional References:

- Core Java, Volume I: Fundamentals Hortsman Pearson 9th 2013
- Core Java, Volume II: Advanced Features Gary Cornell and Hortsman Pearson 8th 2008

- Core Java: An Integrated Approach R. Nageswara Rao DreamTech 1st 2008

Course Code	SEM – IV-Major-1 Practical	Credits	Lectures/Week
24ITMJP41	**Paper I- Core Java Practical**	**2**	**4**

Course Outcomes:

After successful completion of this course, students would be able to

CO1- Define Java classes with appropriate attributes and methods.

CO2- Explain the concepts of data types, control flow, classes, inheritance, exceptions, and event handling, and describe their roles in programming.

CO3- Utilize various object-oriented concepts for problem solving real-life applications.

CO4- Analyze the concepts of thread priority and thread states.

Sr.No.	Topic
1	**Fundamentals of Java Programming:** ● Write a Java program to demonstrate the use of various data types, operators (arithmetic, relational, logical, bitwise), and control flow statements (if-else, switch, for, while, do-while). ● Include examples of type casting (implicit and explicit) and demonstrate the difference between primitive and reference data types.
2	**Classes, Objects, and Constructors:** ● Design a Java class representing a Book with attributes like title, author, ISBN, and price. ● Implement multiple constructors for the Book class: a default constructor, a constructor to initialize all attributes, and a copy constructor. ● Create several Book objects in the main method using different constructors and display their attributes.
3	**Methods and Method Overloading:** ● Enhance the Book class from the previous practical by adding methods like displayDetails(), getPriceWithTax(double taxRate), and isSameBook(Book otherBook). ● Demonstrate method overloading by creating multiple displayDetails() methods with different parameter lists (e.g., one to display all details, another to display only title and author).
4	**Static Members and the Math Class:** ● Create a class Circle with an instance variable radius. Include a static variable PI initialized with the value of pi. ● Implement instance methods to calculate the area and circumference of the circle using the static PI variable. ● Demonstrate the use of static methods (e.g., a static method to calculate the square of a number) within the Circle class.

	• Explore and use various methods from the predefined Math class (e.g., sqrt(), pow(), sin(), cos(), random()).
5	**Inheritance and Method Overriding:** • Design a base class Shape with a method calculateArea() (that might not have a concrete implementation). • Create two derived classes, Rectangle and Triangle, that inherit from Shape and provide their specific implementations for the calculateArea() method (method overriding). • Demonstrate how objects of the derived classes can be treated as objects of the base class (upcasting).
6	**Abstract Classes and Interfaces:** • Define an abstract class GeometricShape with an abstract method getArea(). • Create concrete classes Square and Circle that extend GeometricShape and implement the getArea() method. • Define an interface Colorable with a method setColor(String color) and implement this interface in either Square or Circle (or both). • Demonstrate the use of abstract classes to define a common structure and interfaces to define contracts.
7	**Packages and Access Modifiers:** • Create a package named com.example.shapes. • Move the Shape, Rectangle, Triangle, GeometricShape, Square, and Circle classes from the previous practicals into this package. • Create another class in a different package (e.g., the default package) that uses the classes from the com.example.shapes package. • Demonstrate the use of different access modifiers (public, private, protected, default) and their impact on the visibility and accessibility of classes and members.
8	**Exception Handling:** • Write a Java program that takes two integer inputs from the user and performs division. • Implement try-catch blocks to handle potential ArithmeticException (division by zero) and InputMismatchException (if the user enters non-integer input). • Include a finally block to execute code that should always run, regardless of whether an exception occurred. • Demonstrate how to throw custom exceptions using the throw keyword.
9	**Multithreading:** • Create two threads using the Runnable interface. One thread should print all even numbers from 1 to 50, and the other thread should print all odd numbers from 1 to 50.

	• Implement thread synchronization using the synchronized keyword to ensure that the even and odd numbers are printed in an alternating sequence (or to prevent race conditions if accessing a shared resource). • Explore the use of Thread.sleep(), join(), and yield() methods to control thread execution.
10	**String Manipulation:** • Write a Java program that takes a sentence as input from the user. Perform various string operations such as: 1. Finding the length of the string. 2. Converting the string to uppercase and lowercase. 3. Extracting substrings. 4. Searching for a specific word or character. 5. Replacing characters or substrings. 6. Checking if the string starts or ends with a particular sequence. 7. Splitting the string into words. • Demonstrate the use of various methods from the String and StringBuffer/StringBuilder classes.

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	Major SEM IV	Credits	Lectures /Week
24ITMJ412	Paper II- Embedded System	2	2

Course Outcomes:

After successful completion of this course, students would be able to

CO1-Memorise the basics of computer architecture and microcontrollers.

CO2-Understand the characteristics and quality attributes of embedded systems

CO3-Use different types of sensors for appropriate systems

CO4-Differentiate between general purpose and embedded systems

Unit	Topics	No of Lectures
I	**Introduction:** Basics of computer architecture, computer languages, RISC and CISC architectures, number systems, number format conversions, computer arithmetic, units of memory capacity. **Embedded systems-**Introduction to Embedded System, The hardware point of view Microcontroller unit(MCU), a popular 8-bit MCU, memory for embedded systems, low power design, pull up and pull down resistors **Getting Started with Arduino:** Introduction, Arduino Variants, Install the Drivers, Arduino IDE **Basic Functions:** Overview, Structure, Digital I/O Functions, Analog I/O Functions, Advanced I/O Functions, Timer Functions,Communication Functions, Interrupt Functions, Math Functions, Programming Language Reference	15
II	**Using Sensors with the Arduino:** Light Sensitive Sensors, Temperature Sensors, Temperature and Humidity Sensor, Line-Tracking Sensor, Ultrasonic Sensors, Digital Infrared Motion Sensor, Joystick Module, Gas Sensor and others. **Electromechanical Control Using the Arduino:** DC Motor, Stepper Motor, Servo Motor **Examples of embedded systems:** Mobile phone, automotive electronics, radio frequency identification (RFID), wireless sensor networks(WISENET),Serial Peripheral Device, robotics, biomedical applications, brain machine interface. **Case Studies:** • Air Quality Monitor Using Arduino • A Fire-Fighting Robot Using Arduino • Intelligent Lock System Using Arduino	15

References:

1. Lyla B Das, Embedded systems: An Integrated Approach, 1st Ed., Pearson, 2013 Reference Books:
1. Shibu, K.V., Introduction to Embedded Systems, 1st Ed., TMH, 2009
2. Kanta Rao B, Embedded Systems, 1st Ed., PHI
3. Frank Vahid & Tony Givargis, Embedded System Design, 2nd Edition, John Wiley

Course Code	**Major SEM IV**	**Credit**	**Lectures/Week**
24ITMJP42	**Paper II- Embedded System Practical**	**2**	**4**

Course Outcomes:

After successful completion of this course, students would be able to
CO1- State the embedded application using Embedded C Programming
CO2-Explain the factors involved in selecting the appropriate microcontroller for various applications and describe how Embedded C programming is used in these contexts.
CO3-Apply different types of sensors for appropriate systems
CO4-Design and develop embedded systems using Arduino.

Unit	**Topics**
	Introduction to Arduino
1	Introduction to Arduino circuits and breadboarding Blinking of LEDs
2	Program using Light Sensitive Sensors
3	Program using temperature sensors
4	Programs using humidity sensors
5	Programs using Line tracking sensors
6	Programs using Ultrasonic Sensors
7	Programs using digital infrared motion sensors
8	Programs using gas sensors
9	Programs using servo motors
10	Programs making Joystick with Arduino

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	SEM – VI- Minor	Credits	Lectures /Week
24ITMR421	**Cryptography and Network Security**	**2**	**2**

Course Outcomes:

After successful completion of this course, students would be able to

CO1- Define fundamental terms related to cryptography and network security, such as encryption, decryption, authentication, and key management.

CO2- Explain the principles of encryption and how cryptographic algorithms work.

CO3- Apply cryptographic techniques to secure data transmission and storage.

CO4- Evaluate the effectiveness of different cryptographic algorithms in specific scenarios.

Unit	Topic	No of Lectures
I	Introduction to security attacks - services and mechanism - introduction to cryptography Conventional Encryption: Conventional encryption model - classical encryption techniques substitution ciphers and transposition ciphers – cryptanalysis – steganography - stream and block ciphers - Modern Block Ciphers: Block ciphers principals - Shannon's theory of confusion and diffusion.	**15**
II	Data encryption standard(DES), Principles of public key crypto systems - RSA algorithm - security of RSA - key management – Diffle-Hellman key exchange algorithm - introductory idea of Elliptic curve cryptography	**15**

References:

- W. Stallings, Cryptography and Network Security: Principles and Practice

Additional References:

- Cryptography and Network Security Principles and Practices, Fourth Edition, WilliamStallings, PHI(Pearson)

Course Code	SEM – IV -Minor Practical	Credits	Lectures/ Week
24ITMRP41	**Cryptography Practical**	**2**	**4**

Course Outcomes:

After successful completion of this course, students would be able to

CO1- Recall different types of cryptographic algorithms, including symmetric and asymmetric encryption, hashing, and digital signatures.

CO2- Understand the role of cryptography in securing communication over computer networks.

CO3- Configure and apply network security measures in practical scenarios.

CO4- Evaluate the strengths and weaknesses of various cryptographic algorithms.

Unit	Topic
1	Substitution Techniques a) Write a program to perform substitution ciphers to encrypt the plain text to Caesar cipher and to decrypt it back to plain text. b) Write a program to perform substitution ciphers to encrypt the plain text to ModifiedCaesar cipher and to decrypt it back to plain text.
2	2 Transposition Ciphers: a) Write a program to perform transposition ciphers to encrypt the plain text to cipher and to decrypt it back to plain text using rail fence technique. b) Write a program to encrypt a plain text to a cipher text and decrypt it back to plain text using vernam cipher as the transposition technique
3	Write a program to generate asymmetric Keys for the following Cipher algorithm: DSA (Digital Signature Algorithm).
4	Write a program to generate asymmetric Keys for the following Cipher algorithm: DH (DiffieHellman)
5	Write a program to generate asymmetric Keys for the following Cipher algorithm: RSA.
6	Write a program to encrypt input string by using SecretKey of the following algorithms, and then decrypt the encrypted string and compare the decrypted string with the input string. Use the following algorithms for encryption and decryption: DES
7	Write a program to encrypt input string by using SecretKey of the following algorithms, and then decrypt the encrypted string and compare the decrypted string with the input string. Use the following algorithms for encryption and decryption: BlowFish
8	Write a program to encrypt input string by using SecretKey of the following algorithms, and then decrypt the encrypted string and compare the decrypted string with the input string. Use the following algorithms for encryption and decryption: a.RSA

9	Implement following HashFunctions: RSHash, JSHash, BKDRHash, SDBMHash, DJBHash
10	Write a program to encrypt the given string by using RC4 & MD5 algorithms.

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	OPEN ELECTIVE SEM – IV	Credits	Lectures /Week
24ITOE431	Paper - Probability Distribution	2	2

Course Outcomes:

After successful completion of this course, students would be able to

CO1: Remember the probability distribution concepts

CO2: Understand the basic of standard distribution concepts

CO3: Apply the concepts of hypothesis testing

CO4: Analyse the data using Non-parametric test and Post-hoc analysis of one-way analysis of variance

Unit	Topics	No of Lectures
I	**Standard distributions :** random variable; discrete, continuous, expectation and variance of a random variable, pmf, pdf, cdf, reliability,Introduction and properties without proof for following distributions; binomial,normal, chi-square, t, F. Examples **Hypothesis testing:** one sided, two sided hypothesis, critical region, p-value, tests based on t, Normal and F, confidence intervals.	15
II	**Analysis of variance:** one-way, two-way analysis of variance **Non-parametric tests:** need of non-parametric tests, sign test, Wilicoxon's signed rank test, run test, Kruskal-Walis tests. **Post-hoc analysis of one-way analysis of variance :** Duncan's test Chi-square test of association	15

References:

- Trivedi, K.S.(2009) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

- Ross, S.M. (2006): A First course in probability. 6 th Edn Pearson
- Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. : Common statistical tests.
- Satyajeet Prakashan, Pune(1999)
- Gupta, S.C. and Kapoor, V.K. : Fundamentals of Mathematical Statistics,S. Chand and Sons, New Delhi (2002)
- Gupta, S.C. and Kapoor, V.K. : Applied Statistics, S. Chand and Son's, New Delhi(4 th Edition)
- Montgomery, D.C. : Planning and Analysis of Experiments, Wiley.(2001)

Course Code	SEC SEM – IV	Credits	Lectures/ Week
24ITSE441	Android Mobile Programming Practical	2	4

Course Outcomes:

After successful completion of this course, students would be able to

CO1- Learn the fundamentals of Flutter platform.

CO2- Understand the factors involved in building a cross-platform app.

CO3- Use DART language to build an app.

CO4- Deploy application with single codebase.

Unit	Topics
0	Setting up Flutter, PhoneGAP Project and environment.
1	Program to demonstrate the features of Dart language.
2	Designing the mobile app to implement different widgets.
3	Designing the mobile app to implement different Layouts.
4	Designing the mobile app to implement Gestures.
5	Designing the mobile app to implement the theming and styling.
6	Designing the mobile app to implement the routing.
7	Designing the mobile app to implement the animation.
8	Designing the mobile app to implement the state management.
9	Designing the mobile app working with SQLite Database.
10	Designing the mobile app working with Firebase.

Note: Students are required to complete a minimum of 80% of all the practicals.

Course Code	CEP	Credits
24ITCEP4	Foundations of Community Engagement	2

As per the NEP guidelines, the UG students are expected to complete this program in their Fourth Semester from the academic year 2024-25.

The academic schedule must be planned by the departments, 1 credit to be allotted to classroom and tutorials (15 hours) and 1 credit to field engagement - students learning hours (30 hours)

Classroom Engagement and Field Engagement:

2 credits of classroom engagement and field Engagement comprises of following components:

- Understanding Community Needs
- Identifying Project Opportunities
- Crafting and Finalizing Effective Project Proposals
- Lectures on community sociology and challenges.
- Case studies and discussions on successful community engagement projects.

CEP	**Foundations of Community Engagement**	**[Credits-2]**
Community engagement -Basics (1 Credit)		
Topics Covered	**Activities**	
Introduction to Community Engagement	**Activities** - Overview of theories and models - Importance of interdisciplinary approaches	
Social Issues Analysis	- Guest lecture by a social scientist or experts from diverse sectors - Group discussion and analysis of contemporary social issues	
Community Needs Assessment	-Theory on needs assessment methodologies - Field visit for practical application	

Stakeholder Engagement	- Guest lecture from a community organizer - Simulated stakeholder engagement role-play
Community engagement -Field Work (1 Credit)	
Topics Covered	**Activities**
Cultural Competence in Community Work	- Cultural sensitivity training - Case studies on community engagement
Writing Project Proposal and finance resource management	- Develop a community project proposal and finance resource management - Timeline for implementation
Field Work Skills Training	- Training in data collection, interviewing, and observation - Practical exercises in the community
Ethical Considerations in Community Engagement	- Guest lecture on ethical dilemmas in community work - Case studies and Group discussions

	Credit	**Content/Learning Hours**	**Course Component**
Sem IV	1	15 Hours	Classroom engagement & tutorials
	1	30 (Students learning Hours)	Field Engagement (Requirement Gathering)

Note: Class engagement: 1 Credit = 15 Hour

For field engagement/ Field Project: 1 Credit = 30 Hours

Hours Evaluation of Classroom Engagement and Field Engagement (Sem. IV) Evaluate each student for 50 marks per

- Semester at department level -
  - 20 marks for Continuous evaluation (CE)

- Participation in class activities and discussions.
- Submission of reflective essays.

- 30 marks for End Semester Examination (ESE)

➢ Based on evaluation of Project Proposal.

UGC Recommended field-based activities:

1. Interaction with Self Help Groups (SHGS) women members, and study their functions and challenges, planning for their skill-building and livelihood activities.
2. Visit Mahatma Gandhi National. Rural Employment Guarantee Act 2005 (MGNREGS)project sites, interact with beneficiaries and interview functionaries at the work site.
3. Field visit to Swachh Bharat project sites, conduct analysis and initiate problem solving measures.
4. Conduct Mission Antyodaya surveys to support under Gram Panchayat Development Plan (GPDP) etc.Interactive community exercise with local leaders, panchayat functionaries, grass-root officials and local institutions regarding village development plan preparation and resource mobilization.
5. Visit Rural Schools/mid-day meal centres, study academic and infrastructural resources, digital divide and gaps;
6. Participate in Gram Sabha meetings, and study community participation;
7. Associate with Social audit exercises at the Gram Panchayat level, and interact with program beneficiaries;
8. Visit to local Nagarpalika office and review schemes for urban informal workers and migrants;
9. Attend Parent Teacher Association meetings, and interview school dropouts; Visit local Anganwadi Centre and observe the services being provided
10. Visit local NGOs, civil society organisations and interact with their staff and beneficiaries;
11. Organize awareness programs, health camps, Disability camps and cleanliness
12. camps;
13. Conduct soil health test, drinking water analysis, energy use and fuel efficiency surveys and building solar powered village;

14. Raise understanding of people's impacts of climate change, building up community's disaster preparedness;
15. Organize orientation programs for farmers regarding organic cultivation, rational use of irrigation and fertilizers, promotion of traditional species of crops and plants and awareness against stubble burning;
16. Formation of committees for common property resource management, village pond maintenance and fishing;
17. Identifying the small business ideas (handloom, handicraft, khadi, food products, etc.) for rural areas to make the people self-reliant.
18. Any other Community engagement activity with approval of BOS and Academic Council.

(Note that every department can also find CEP allied with their subject.)

Reference:

1. Guidelines on "Fostering Social Responsibility & Community Engagement in HigherEducation Institutions in India 2.0 (https://www.ugc.gov.in/publication/ebook)

Evaluation Scheme for Second Year (UG) under NEP (2 credits)

I. Internal Evaluation for Theory Courses – 20 Marks

1) Continuous Internal Assessment(CIA) Assignment - Tutorial/ Case Study/ Project / Presentations/ Group Discussion / Ind. Visit. – 10 marks

2) Continuous Internal Assessment(CIA) ONLINE Unit Test – 10 marks

II. External Examination for Theory Courses – 30 Marks

Duration: 1 Hours

Theory question paper pattern: All questions are compulsory.

Question	Based on	Marks
Q.1	Unit I	15
Q.2	Unit II	15

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & the allocation of Marks depends on the weightage of the topic.

Paper Pattern of Theory Paper:

DES's Kirti M. Doongursee College (AUTONOMOUS), Dadar (W), Mumbai-28			
Regular / Additional / ATKT Examination			
Duration: 1 Hour			**Max Marks: 30**
Date: **Time:** **Code:** ***(For office use)***			
N. B.	i)	*All Questions are compulsory.*	
	ii)	*Mixing of sub-questions is not allowed*	
	iii)	*Draw neat labeled diagrams wherever necessary.*	
Q. No.			**Marks**
Q.1 A)			**05**
OR			
Q.1 B)			**05**
Q.1 C)			**05**
OR			
Q.1 D)			**05**
Q.1 E)			**05**
OR			
Q.1 F)			**05**
Q.2 G)			**05**
OR			
Q.2 H)			**05**

Q.2 I)		**08**
OR		
Q.2 J)		**08**
Q.2 K)		**02**
OR		
Q.2 L)		**02**

III. Practical Examination

- Each core subject carries 50 Marks.
- Duration: 2 Hours for each practical course.
- Minimum 80% practical from each core subjects are required to be completed.
- Certified Journal is compulsory for appearing at the time of Practical Exam

NOTE: To pass the examination, attendance is compulsory in both Internal & External (Theory + Practical) Examinations.