AC: 02.06.2025 ITEM NO: 1.2

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: Computer Science

Choice Based Credit System (CBCS)
with effect from
Academic Year 2025-2026

# PROGRAM OUTCOMES

| **PO** | **Description** |
|---|---|
| A student completing Bachelor's Degree in **Science** Program will be able to | |
| PO1 | Disciplinary Knowledge:<br>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:<br>Exhibit the skills of analysis, inference, interpretation and problem-solving by observing the situation closely and design the solutions |
| PO3 | Social competence:<br>Display the understanding, behavioral skills needed for successful social adaptation, work in groups, exhibit thoughts and ideas effectively in writing and orally. |
| PO4 | Research-related skills and Scientific temper:<br>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:<br>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:<br>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:<br>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:<br>Understand the impact of the scientific solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development. |

## Year of implementation- 2025-2026

## Name of the Department-Computer Science

| Semester | Course Code | Course Title | Vertical | Credit |
|---|---|---|---|---|
| **III** | 25CSMJ311 | Data Structure | Major | 2 |
| | 25CSMJP31 | Data Structure Practical | Major | 2 |
| | 24CSMJ312 | Advance DBMS | Major | 2 |
| | 24CSMJP32 | Advance DBMS Practical | Major | 2 |
| | 24CSMR321 | Java Application Development | Minor | 2 |
| | 24CSMRP31 | Java Application Development Practical | Minor | 2 |
| | 24CSOE331 | Research Methodology-I | OE | 2 |
| | 24CSVC341 | Advance Application Development | VSC | 2 |
| | 24CSFP3 | Field Project | Field Project | 2 |
| **IV** | 24CSMJ411 | Theory of Computation | Major | 2 |
| | 24CSMJP41 | Theory of Computation Practical | Major | 2 |
| | 24CSMJ412 | Software Engineering | Major | 2 |
| | 24CSMJP42 | Software Engineering Practical | Major | 2 |
| | 24CSMR421 | Internet of Things | Minor | 2 |
| | 24CSMRP41 | Internet of Things Practical | Minor | 2 |
| | 24CSOE431 | Research Methodology-II | OE | 2 |
| | 24CSSE441 | Android Application Development | SEC | 2 |
| | 24CSCP4 | Community Engagement Program | CEP | 2 |

# SEMESTER-III

| Course Code | MAJOR SEM – III | Credits | Lectures/Week |
|---|---|---|---|
| 25CSMJ311 | Paper I - Data Structure | 2 | 2 |

**Course Outcomes:**

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

CO1: Learn and describe elementary data structures such as arrays,stack, queue, linked list, tree, graphs and Algorithms

CO2: Understand the various concepts related to different data structures and Algorithms

CO3: Design and write programs for elementary data structures such as stack, queue, linked list. Write algorithms / programs for solving problems with the help of fundamental data structures trees, graphs.

CO4: Analyze use of different data structures and algorithms on different types of problems.

| Unit | Topics | No of Lectures |
|---|---|---|
| I | **Introduction:** Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types<br>**Algorithm**<br>Simple Data Structures:<br>**Array:** Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Multi-dimensional Arrays, Memory Representation of Two Dimensional Arrays, Advantages and Limitations of Arrays.<br>**Stack:** Introduction, Operations on the Stack, Memory Representation of Stack, Array Representation of Stack, Recursion.<br>**Queue:** Introduction, Operations on the Queue, Memory Representation of Queue, Array representation of Queue, Circular Queue, Priority Queue<br>**Linked List**: Linked List, singly Linked List, Traversal of Linked List, Searching, Insertion in Linked List, Deletion from Linked List,Circular Linked List, Doubly Linked List, Traversal, Search, Insert and Delete in a doubly Linked List, Implementing other Data Structures. | 15 |
| II | **Trees:** Tree terminologies, Binary Tree, Properties of Binary Tree, Memory Representations of Binary Tree - static and dynamic , Binary Tree Traversals, Binary Search Tree, Operations on Binary Search Tree- search, insert, traversal<br>Graphs: Terminologies, graph representations, shortest path | 15 |

**References:**

- Seymour Lipschutz,Schaum’s Outlines Data structure Tata McGraw Hill 2nd 2005
- AM Tanenbaum, Y Langsam and MJ Augustein,Data structure – A Pseudocode Approach with C, Prentice Hall India 2nd 2006

**Additional References:**

- Lalit Goyal, Vishal Goyal, Pawan Kumar,A Simplified Approach to Data Structures SPD 1 st 2014
- Jean – Paul Tremblay and Paul Sorenson, An Introduction to Data Structure with Applications Tata MacGraw Hill 2 nd 2007
- Maria Rukadikar,Data Structure and Algorithm SPD 1 st 2017

| Course Code | MAJOR SEM-III Practical | Credits | Lectures/Week |
|---|---|---|---|
| **25CSMJP31** | **CS Practical 311 Data Structure** | **2** | **4** |

**Course Outcomes:**

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

CO1: List the errors and warnings for the given input.

CO2: Understand / demonstrate the concepts, design and execution process of the programs.

CO3: Write a modularized program for implementing various data structures and algorithm

CO4: Compare and use different concepts of algorithms and data structures to solve given problems.

| | |
|---|---|
| **1** | Writing simple programs using one dimensional and two dimensional array |
| **2** | Write a program to implement a stack using an array. |
| **3** | Write a program: factorial and Fibonacci series using recursion |
| **4** | Write a program to implement a linear queue |
| **5** | Write a program to implement a circular queue |
| **6** | Write a program to implement a singly linked list |
| **7** | Write a program to implement a doubly linked list |
| **8** | Write a program to implement a stack using a linked list. |
| **9** | Write a program to create and traverse a binary search tree |
| **10** | Write a program to construct and print adjacency matrix representation of a graph. |

| Course Code | MAJOR SEM – III | Credits | Lectures /Week |
|---|---|---|---|
| 24CSMJ312 | Paper II - Advance DBMS | 2 | 2 |

**Course Outcomes:**

CO1: Remember the concepts of Database Programming Paradigms.
CO2: Understanding the various Concepts of PL/SQL
CO3: Apply and implement different features of PL/SQL to database
CO4: Analyze different techniques of PL/SQL (stored procedure, functions, cursors , trigger)

| Unit | Topics | No of Lectures |
|---|---|---|
| I | **Introduction:** Defining variables and constants,PL/SQL expressions and comparisons: Logical Operators, Boolean Expressions, CASE Expressions Handling, Null Values in Comparisons and Conditional Statements, PL/SQL Datatypes: Number Types, Character Types, Boolean Type, Date time and Interval Types. The %TYPE Attribute ,The %ROWTYPE Attribute<br>Overview of PL/SQL Control Structures: Conditional Control: IF and CASE Statements, IF-THEN Statement, IF-THEN-ELSE Statement, IF THEN-ELSEIF Statement, CASE Statement,<br>Iterative Control: LOOP and EXIT Statements, WHILE-LOOP, FOR-LOOP, Sequential Control: GOTO and NULL Statements, Continue<br>**Sequences:** creating sequences, referencing, altering and dropping a sequence | 15 |
| II | **Stored Procedures & Functions:** Types and benefits of stored procedures, creating stored procedures, executing stored procedures, altering stored procedures, viewing stored procedures. Create a Simple Function, Execute a Simple Function, recursive function.<br>**Triggers:** Concept of triggers, Implementing triggers – creating triggers, Insert, delete, and update triggers, viewing, deleting and modifying triggers, and enforcing data integrity through triggers. Ex<br>**Cursors:** Concept of a cursor, types of cursors: implicit cursors;Explicit cursor, Cursor for loops, Cursor variables, parameterized cursors,nested cursors, FOR UPDATE Clause and WHERE CURRENT Clause,<br>**Indexing**: create, modify and drop index | 15 |

**References:**

- Ivan Bayross, “SQL,PL/SQL -The Programming language of Oracle”, B.P.B. Publications
- Michael Abbey, Michael J. Corey, Ian Abramson, Oracle 8i – A Beginner’s Guide, TataMcGraw-Hill, 3rd edition
- Sheila Moore, E. Belden,PL/SQL Language Reference 11g, 2nd edition.
- Ramakrishnam, Gehrke, “Database Management Systems”, McGraw- Hill, 3rd edition.

**Additional References:**

- Robert Sheldon, Geoff Moes, Beginning MySQL, Wrox Press
- Joel Murach, Murach’s MySQL, Murach

| **Course Code** | **MAJOR SEM-III Practical** | **Credits** | **Lectures/Week** |
|---|---|---|---|
| **24CSMJP32** | **CS Practical 312 Advance DBMS** | **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 DML, DDL.

CO2: Understand and demonstrate the basic programming construct of PLSQL

CO3: Write a modularized program for implementing various concepts of PLSQL.

CO4: Evaluate the performance of different methods in specific database scenarios.

| | |
|---|---|
| **1** | Writing PL/SQL Blocks with basic programming constructs by including following:<br>a. Sequential Statements<br>b. unconstrained loop |
| **2** | Writing PL/SQL Blocks with basic programming constructs by including following:<br>a. If…then…Else, IF…ELSEIF…ELSE… END IF<br>b. Case statement |
| **3** | Writing PL/SQL Blocks with basic programming constructs by including following:<br>a. While-loop Statements<br>b. For-loop Statements<br>c. Unconstrained loops |

| | |
|---|---|
| **4** | Writing PL/SQL Blocks with basic programming constructs by including Sequences:<br>a. Creating simple Sequences with clauses like START WITH, INCREMENT<br>BY, MAXVALUE, MINVALUE, CYCLE \| NOCYCLE, CACHE \| NOCACHE, ORDER \| NOORECER.<br>b. Creating and using Sequences for tables. |
| **5** | Writing Procedures in PL/SQL Block (IN, OUT, INOUT, DEFAULT keywords).<br>a. Create an empty procedure, replace a procedure and call procedure<br>b. Create a stored procedure and call it<br>c. Define procedure to insert data<br>d. A forward declaration of procedure |
| **6** | Writing Functions in PL/SQL Block.<br>a. Define and call a function<br>b. Define and use function in select clause,<br>c. Call function in dbms_output.put_line<br>d. Recursive function<br>e. Count Employee from a function and return value back<br>f. Call function and store the return value to a variable |
| **7** | Creating and working with Insert/Update/Delete Trigger using Before/After clause. |
| **8** | Write an Implicit and explicit cursor to complete the task. |
| **9** | Create packages and use it in SQL black to complete the task. |
| **10** | Create Index, modify index and drop index |

| Course Code | MINOR SEM – III | Credits | Lectures/Week |
|---|---|---|---|
| **24CSMR321** | **Java Based Application** | **2** | **2** |

**Course Outcomes:**

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

CO1: Introduces Object Oriented Programming concepts using the Java language

CO2: Understand the concepts and features of object oriented programming

CO3: Apply object oriented programming features and concepts for solving the given problem

CO4: Analyze different features and concepts of oops.

| Unit | Topics | No of Lectures |
|---|---|---|
| **I** | **OOPS:**Introduction to java, Class, Object, Static Keywords, Constructors, this keyword, Inheritance, Inner class, Anonymous Inner class, super keyword, Polymorphism (overloading and overriding), Abstraction, Encapsulation, Abstract Classes, Interfaces.<br>**Packages:** Introduction to predefined packages, User Defined Packages, Access specifiers<br>**Exception Handling**: Introduction, Pre-Defined Exceptions, try-catch-finally, throws, throw, User Defined Exceptions<br>**Multithreading:** Thread Creations, Thread Life Cycle, Life Cycle Methods, Synchronization, wait() notify() notify all() methods | **15** |
| **II** | **Introduction to JFC and Swing**: Features of the Java Foundation Classes, Swing API Components, JComponent Class, Windows, Dialog Boxes, and Panels, Labels, Buttons, Check Boxes, Menus, Toolbars, Implementing Action interface, Pane, JScrollPane, Desktop pane, Scrollbars, Lists and Combo Boxes, Text-Entry Components.<br>**Event Handling:** Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes.<br>**JDBC:** Introduction, JDBC Architecture, JDBC Drivers, JDBC Connectivity Model, java.sql package, Using Statement, PreparedStatement, CallableStatement, ResultSet, Scrollable and Updatable ResultSet, Navigating and manipulating data, ResultSetMetaData, Managing Transactions in JDBC, JDBC Exception classes, BLOB & CLOB | **15** |

**References:**

- Herbert Schildt, Java The Complete Reference, Eleventh Edition, McGraw-Hill Education, 2020
- Bryan Basham, Kathy Sierra, Bert Bates, Head First Servlets and JSP, O’reilly (SPD), 2018
- Cay S. Horstmann, Gary Cornell, Core Java™ 2: Volume II–Advanced Features Prentice Hall PTR, 2004

| **Course Code** | **MINOR SEM-III Practical** | **Credits** | **Lectures/Week** |
|---|---|---|---|
| **24CSMRP32** | **CS Practical Java Based Application** | **2** | **4** |

**Course Outcomes:**
After successful completion of this course, students would be able to

CO1: Remember and describe different concepts of Constructor,Inheritance, Swing, Exception handling,Event Handling

CO2: Understand the concept and underlying principles of Object-Oriented Programming.

CO3: Apply problem-solving and programming skills using the OOP concept.

CO4: Analyze the concepts of the oops to solve real-world problems.

| | |
|---|---|
| **1** | a. Write a program to create a class and implement the concepts of Constructor Overloading, Method Overloading, Static methods<br>b. Write a program to implement the concept of Inheritance and Method Overriding |
| **2** | a. Write a program to implement the concepts of Abstract classes and methods<br>b. Write a program to implement the concept of interfaces |
| **3** | Write a program to define user defined exceptions and raise them as per the requirements. |
| **4** | Write a program using various swing components to design Java applications to accept a student's resume. (Design form) |
| **5** | Write a JDBC program that displays the data of a given table |
| **6** | Write a JDBC program to insert / delete records into a given table |
| **7** | Construct a simple calculator using the JAVA Swings with minimum functionality. |
| **8** | Construct a GUI using JAVA Swings to accept details of a record of a given table and submit it to the database using JDBC technology on the click of a button. |
| **9** | Construct a simple Login using the JAVA Swings with minimum functionality. |
| **10** | Construct a simple Calculator using the JAVA Swings with minimum functionality |

| Course Code | OPEN ELECTIVE SEM – III | Credits | Lectures /Week |
|---|---|---|---|
| **24CSOE331** | **Research Methodology- I** | **2** | **2** |

**Course Outcomes:**

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

CO1: Remember Types of research, definition.

CO2: Understand the format of research design, its steps and Various types for financial decision making.

CO3: Apply various concepts of research in decision making/ to be able to handle a small survey.

CO4 : Analyze existing secondary data and Research

| Unit | Topics | No of Lectures |
|---|---|---|
| **I** | **Introduction to Research**- Introduction and meaning of research, Objectives of research, Features and Importance of research in Accounting and Finance, Objectives and Types of research - Basic, Applied, Descriptive, Analytical and Empirical Research. Formulation of research problem : Meaning and Selection Review of Literature | **15** |
| **II** | **Research Design** Meaning of Introduction, Need, and Good research design.<br>Hypothesis: Formulation, Sources, Importance and Types Different Research designs<br>Data Collection: Introduction and meaning, types of data Primary data: Observation, Experimentation, Interview, Schedules, Survey, Questionnaires, Limitations of Primary data Secondary data: Sources and Limitations Factors affecting the choice of method of data collection. | **15** |

**References:**

- Malcolm Smith, Research Methods in Accounting, Fourth edition, 2017 SAGE Publications Ltd.
- Viv Beattie and Bob, Research Methods and Methodology in Finance and Accounting, Ryan, 2 nd Edition 2002 Cengage Learning EMEA.

| Course Code | VOCATIONAL SKILL COURSE SEM – III | Credits | Lectures/Week |
|---|---|---|---|
| **24CSVC341** | **Paper I -** Advance Application Development<br>**(Practical Based)** | **2** | **4** |

**Course Outcomes:**

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

CO1: Learn MEANstack structure and its components.

CO2: Understand Meanstack technologies such as MongoDB, Express.js, AngularJS, and Node.js.

CO3: Develop  Meanstack Framework for building fast and scalable applications.

CO4: Analyze and integrate the front-end and back-end components of the MEAN stack.

| Unit | Topics | No of Sessions |
|---|---|---|
| **I** | **Node.js(N):** Introduction to Node.js. Installing Node.js. The package.json File. The Node.js Event Loop. The I/O Cycle. The Anatomy of a Node.js Module. Creating Node Modules. Exploring the Node.js HTTP Module. Creating an HTTP Web Server with Node.js. Responding to HTTP Requests. Routing in Node.js. Creating a Sample Node.js Application.<br>**MongoDB(M):** Introduction to MongoDB. Installing MongoDB. Using MongoDB Compass. Using Mongo Shell Interface. Connecting to MongoDB. Creating Schemas and Models. Querying Documents Using find(). Inserting Documents Using create(). Updating Documents Using findOneAndUpdate(). Deleting Documents Using findOneAndDelete() & deleteMany() | **15** |
| **II** | **Server-Side Development with Express (E):** Introduction to the Express Framework. Installing and Testing Express. Creating a Node.js Express App. Restructuring an Express App. Creating Templates. Using Express Middleware Functions. Creating the List Page. Creating the Details Page. Creating the Edit Page. Creating the Add Page. Deleting Data. REST API Basics. Testing REST APIs. Refactoring APIs.<br>**Understanding Angular.JS(A):** Getting Started with Angular. Creating an Angular Application. Angular Project File Structure. Anatomy of an Angular Component. One-way Data Binding. Two-way Data Binding. Using Nglf Directive. Using NgForOf Directive. Angular Modules. Creating NgModules Using Angular Router. Configuring Templates. Creating Navigations. Working with Template-driven Forms. Working with Reactive Forms. Validating Form Data. Services Dependency Injection (DI). Reading Data | **15** |

| | | |
|---|---|---|
| | from Database. Inserting Data into Database. Updating Data in the Database. Delete Data from Database. | |

**References:**

- Brad Dayley, Brendan Dayley, Caleb Dayley, Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications Pearson, 2018.
- Marco L. Napoli, Beginning Flutter: A Hands On Guide to App Development, Wrox, 2019

**Additional References:**

- Adam Bretz, Colin J Ihrig, Shroff,Full Stack Javascript Development with Mean - MongoDB, Express, AngularJS, and Node.JS,SitePoint, 2015
- Zammetti Frank, Practical Flutter,, Apress, 2019

| Course Code | VSC SEM-III Practical | Credits | Lectures/ Week |
|---|---|---|---|
| **24CSSC341** | Advanced Application Development Practical | 2 | 4 |

**Course Outcomes:**

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

CO1: To learn application development components
CO2: To understand the necessary and important technologies such as MongoDB, Express.js, AngularJS, and Node.js
CO3: Develop  dynamic application framework using MEANstack.

CO4: Analyze and integrate the components of the MEAN stack.

| Unit | Topics |
|---|---|
| **1** | Write a program to implement following concepts in Node.js<br>- Event loop<br>- modules<br>- Creating an HTTP Web Server with Node.js.<br>- Responding to HTTP Requests.<br>- Routing in Node.js.<br>- Creating a Sample Node.js Application. |
| **2** | NodeJS<br>1.Nodejs execution on Console<br>- Write a program to perform -addition, subtraction, multiplication and division<br>- Write a program using different string functions<br>- Write a program using typeOf function to handle the data.<br>- Write a program to create a class Company and display it's object value.<br>- Write a program to create sever and display your Name.<br>2.handling REPL environment |

| | |
|---|---|
| | (addition, subtraction, division, multiplication, power, squareroot, round, floor, ceil) |
| **3** | Write a program to create custom middleware in Express. |
| **4** | Express installation<br>-Write a program to print hello world<br>-Write a program to display current time<br>-Write a program to create a route and display hello world<br>-Write a program to create function and route<br>-Write a program to create a route and send parameters<br>-Write a program to create route and matches the parameter pattern |
| **5** | Write a program to implement CRUD operations on MongoDB**.** |
| **6** | AngularJS<br>1. Write a program to design an HTML page with two fields- FirstName and Lastname. Apply data binding on the same fields.<br>2 Write a program to perform the task- addition, subtraction, division and multiplication using an evaluation tag.<br>3. Write a program to design a controller.<br>4. Write a program to design registration form and add data binding and controller to it |
| **7** | AngularJS<br>1.Write a program to design html form (Firstname, Lastname, DOB, Contact No) and apply the filters (uppercase,lowercase,date and number) on each input.<br>2. Write a program to display a list and sort it using Order by filter<br>3. Write a program to create a custom filter |
| **8** | Design Login form using Node JS and MongoDB |
| **9** | Design Feedback form using Pugjs, Node.js, MongoDB, Express |
| **10** | Design Website using Mean Stack architecture |

| Course Code | FP | Credits | HOURS |
|---|---|---|---|
| **24CSFP351** | **Field Project** | **2** | **60** |

### Field Project Implementation 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.
Course outcomes should be designed at departmental level in alignment with above course objectives

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

Note: Field Project 1 Credit = 2 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. (Field diary)

-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 | MAJOR SEM – IV | Credits | Lectures /Week |
|---|---|---|---|
| 24CSMJ411 | Paper I- Theory of Computation | 2 | 2 |

**Course Outcomes:**

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

CO1: Remember and describe different concepts of finite automata,regular expressions, grammars, PDA,LBA and Turing machines

CO2: Understand and discuss the concepts, properties, types of automata, regular expressions, grammars, PDA and Turing machines.

CO3: Construct different automata and regular expressions and minimize a DFA. Construct CFG, simplify CFG,Represent CFG and construct a PDA and a Turing machine

CO4: Compare and analyze different languages and their representations, properties, acceptor machines

| Unit | Topics | No of Lectures |
|---|---|---|
| I | **Automata Theory:** Introduction, Defining Automaton, Finite Automaton, Transitions and Its properties, Acceptability by Finite Automaton, Nondeterministic Finite State Machines, Mealy and Moore Machines, Minimizing Automata.<br>**Regular Sets:** Regular Expressions, Finite automata and Regular Expressions, Pumping Lemma and its Applications, Closure Properties. | 15 |
| II | **Formal Languages:** Defining Grammar, Derivations, Languages generated by Grammar, Chomsky Classification of Grammar and Languages<br>**Context-free Languages:** Derivation Tree,Ambiguous Grammar, CFG simplification, Normal Forms, PDA<br>**Turing Machines:** Concept and problems | 15 |

**References:**

- K. L. P Mishra, Chandrasekharan, Theory of Computer Science, PHI,3rd Edition 2019

**Additional References:**

- John E. Hopcroft,Introduction to Automata Theory, Languages and Computation, Pearson Education, 2014
- Daniel Cohen,Introduction to Computer Theory, Wiley, 2nd Edition, 2007
- E.V. Krishnamurthy,Introductory Theory of Computer Science, Affiliated East-West Press, 2009

| Course Code | MAJOR SEM-IV Practical | Credits | Lectures/Week |
|---|---|---|---|
| **24CSMJP41** | **CS Practical 411 Theory of Computation** | **2** | **4** |

**Course Outcomes:**

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

CO1: List the errors and warnings for the given input.

CO2: Explain various concepts and demonstrate the execution process of the programs.

CO3: Write modularized program code for implementing different concepts

CO4: Compare and analyze use of different concepts to solve different types of problems

| | |
|---|---|
| **1** | Write a program for tokenization of given input |
| **2** | Design a Program for creating a machine that accepts three consecutive 1. |
| **3** | Design a Program for creating a machine that accepts the string always ending with 101. |
| **4** | Design a program for accepting decimal numbers divisible by 2. |
| **5** | Write a program for generating language strings for given regular expression 1(0+1)* 0 |
| **6** | Write a program for generating language strings for given regular expression (0+1)* 10 (0+1)* |
| **7** | Write a program for generating derivation sequence / language for the given sequence of productions |
| **8** | Design a program for creating a machine which accepts string having equal no. of a's and b's starting with all a's. |
| **9** | Design a PDA that accepts the language an bn where n>0 |
| **10** | Design a PDA to accept WCWR where W is any string and WR is the reverse of the string W and C is a Special symbol. |

| Course Code | MAJOR SEM – II | Credits | Lectures /Week |
|---|---|---|---|
| 24CSMJ412 | Paper II - Software Engineering | 2 | 2 |

**Course Outcomes:**

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

CO1: Learn different software engineering process life cycle models and concepts.

CO2: Understand the techniques of system validation and evaluation

CO3: Apply system modeling concepts necessary for software project development.

CO4: Analyze and translate a specification into a design, and then realize that design practically, using an appropriate software engineering methodology.

| Unit | Topics | No of Lectures |
|---|---|---|
| I | **Introduction:** The Nature of Software, Software Engineering, Professional Software Development, Layered Technology, Process framework, CMM, Process Patterns and Assessment Prescriptive Models: Waterfall Model, Incremental, RAD Models Evolutionary Process Models: Prototyping, Spiral and Concurrent Development Model Specialized Models: Component based, Aspect Oriented development, The Unified Process Phases, Agile Development-Agility, Agile Process, Extreme Programming<br>**Requirement Analysis and System Modeling:** Requirements Engineering, Eliciting Requirements, SRS Validation, Components of SRS, Characteristics of SRS, Object-oriented design 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 | **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<br>**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<br>**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 | 15 |

|  |  |  |
|---|---|---|
|  | **Project Scheduling:** Basic Principles, Relationship Between People and Effort, Effort Distribution, Time-Line Charts |  |

**References:**

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

**Additional References:**

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

| **Course Code** | **MAJOR SEM-IV Practical** | **Credits** | **Lectures/Week** |
|---|---|---|---|
| 24CSMJP42 | CS Practical 412 **Software Engineering** | 2 | 4 |

**Course Outcomes:**
After successful completion of this course, students would be able to
CO1: Gain knowledge about implementing software engineering methods.
CO2: Understand the software engineering methodologies involved in the phases for project development.
CO3: Apply software engineering techniques in system development and develop a prototype.
CO4: Analyze different case studies for implementing OO concepts.

|  |  |
|---|---|
| **1** | Write down the problem statement for a suggested system of relevance. |
| **2** | Perform requirement analysis and develop Software Requirement Specification Sheet (SRS) for suggested systems. |
| **3** | Draw the function oriented diagram: Event Table, Data Flow Diagram (DFD) and Structured chart. |
| **4** | Draw the user's view analysis for the suggested system: Use case diagram. |
| **5** | Draw the structural view diagram for the system: Class diagram, object diagram. |
| **6** | Draw the behavioral view diagram : State-chart diagram, Activity diagram. |
| **7** | Draw the behavioral view diagram for the suggested system: Sequence diagram, Collaboration diagram. |
| **8** | Draw the implementation and environmental view diagram: Component diagram, Deployment diagram. |
| **9** | Perform Estimation of effort using FP Estimation. |
| **10** | Build a prototype of a case study and Prepare timeline chart/Gantt Chart/PERT Chart. |

| Course Code | MINOR SEM – IV | Credits | Lectures /Week |
|---|---|---|---|
| 24CSMR421 | Paper I - Internet of Things | 2 | 2 |

**Course Outcomes:**

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

CO1- Learn different protocols used for IoT design
CO2- Understand roles of sensors in IoT
CO3- Use sensors and actuators for the design of IoT.
CO4- Analyze the different roles of big data, cloud computing and data analytics in a typical IoT system.

| Unit | Topics | No of Lectures |
|---|---|---|
| I | **Introduction to IOT:** Understanding IoT fundamentals, IOT Architecture and protocols, Various Platforms for IoT, Real time Examples of IoT, Overview of IoT components and IoT Communication Technologies, Challenges in IOT.<br>**Arduino Simulation Environment:** Arduino Uno Architecture ,Setup the IDE, Writing Arduino Software, Arduino Libraries, Basics of Embedded C programming for Arduino, Interfacing LED, push button and buzzer with Arduino, Interfacing Arduino with LCD.<br>**Sensor & Actuators with Arduino:** Overview of Sensors working, Analog and Digital Sensors, Interfacing of Temperature, Humidity, Motion, Light and Gas Sensor with Arduino, Interfacing of Actuators with Arduino. Interfacing of Relay Switch and Servo Motor with Arduino. | |
| II | **Basic Networking with ESP8266 WiFi module**: Basics of Wireless Networking, Introduction to ESP8266 Wi-Fi Module, Various Wi-Fi library, Web server- introduction, installation, configuration Posting sensor(s) data to web server.<br>**IoT Protocols**: M2M vs. IOT, Communication Protocols.<br>**Cloud Platforms for IOT**: Virtualization concepts and Cloud Architecture, Cloud computing, benefits, Cloud services -- SaaS, PaaS, IaaS, Cloud providers & offerings, Study of IOT Cloud platforms, ThingSpeak API and MQTT, Interfacing ESP8266 with Web services. | |

**References:**

- Hakima Chaouchi, — “The Internet of Things Connecting Objects to the Web” ISBN : 978-1- 84821-140-7, Wiley Publications
- Olivier Hersent, David Boswarthick, and Omar Elloumi, — “The Internet of Things: Key Applications and Protocols”, WileyPublications
- Vijay Madisetti and ArshdeepBahga, — “Internet of Things (A Hands-on-Approach)”, 1 st Edition, VPT, 2014.

- J. Biron and J. Follett, "Foundational Elements of an IoT Solution", O'Reilly Media, 2016.
- Keysight Technologies, “The Internet of Things: Enabling Technologies and Solutions for Design and Test”, Application Note, 2016.

**Additional References:**

- Daniel Minoli, — “Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”, ISBN: 978-1-118-47347-4, Willy Publications
- Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press

| **Course Code** | **MAJOR SEM-IV Practical** | **Credits** | **Lectures/ Week** |
|---|---|---|---|
| **24CSMJP43** | **CS Practical 413 Internet Of Things** | 2 | 4 |

**Course Outcomes:**

After successful completion of this course, students would be able to
CO1- Learn the fundamentals of IoT and Arduino Uno architecture.

CO2- Choose the right microcontroller and C Programming for various Applications.

CO3- Design and develop embedded systems using Arduino.

CO4- Analyze different types of sensors for appropriate circuit design.

| **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 |

| **Course Code** | **OPEN ELECTIVE SEM – IV** | **Credits** | **Lectures /Week** |
|---|---|---|---|
| **24CSOE431** | **Research Methodology- II** | **2** | **2** |

**Course Outcomes:**

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

CO1: Remember Methods of data collection, Types of report format

CO2:Understand Statistical Analysis: Tools and Techniques, Correlation & Regression Analysis.

CO3: Apply data collection methods for collecting data and methods of various report writing.

CO4 : Analyze Significance, Data Presentation and Types of research

| **Unit** | **Topics** | **No of Lectures** |
|---|---|---|
| **I** | Data Collection and Processing:<br>Factors affecting the choice of method of data collection. Sampling: Significance, Methods, Factors determining sample size.Data Presentation: Significance in Research, Stages in Data Processing: Editing, Coding, Classification, Tabulation, Graphic Presentation.Statistical Analysis: Tools and Techniques, Measures of Central Tendency, Measures of Dispersion, Correlation Analysis and Regression Analysis.Use of computer and internet in data collection and processing | **15** |
| **II** | Interpretation and Report Writing:<br>Meaning and techniques of interpretation, Research Report Writing: Importance,Essentials, Structure/ layout, Types. | **15** |

**References:**

- Malcolm Smith, Research Methods in Accounting, Fourth edition, 2017 SAGE Publications Ltd.
- Viv Beattie and Bob, Research Methods and Methodology in Finance and Accounting, Ryan, 2 nd Edition 2002 Cengage Learning EMEA.

| **Course Code** | **SKILL ENHANCEMENT COURSE**<br>**SEM – IV** | **Credits** | **Lectures /Week** |
|---|---|---|---|
| **24CSSC451** | **Paper I -** Android Application Development | **2** | **2** |

**Course Outcomes:**

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

CO1: Learn fundamental concepts of Android.

CO2: Understand Android OS, grade, Android Studio.

CO3: Apply Kotlin programming concepts to Android application development

CO4: Analyze Data Storage-Shared Preference,SQLite Databases and Firebase Real-Time Data

| **Unit** | **Topics** | **No of Lectures** |
|---|---|---|
| **I** | **App Development with Android Studio**: Android Architecture, Android Application Framework, Android Virtual Device, Creating and running First Android Application, working with Physical Android Device, Adding Kotlin Files in Android Studio<br>**Basics Of Android**- Application Components: Activities, Intent, and Broadcast Receiver, Services, Fragment, Activity Life Cycle, Content Provider, Widgets, and Notifications | **15** |
| **II** | **Designing Android UI**: User Interface (UI), Layout and Its Types, Layout Attribute, working with Views, Android UI Controls, Styles and Themes, Event Handler, setting up themes in Manifest and from the application, dialog in activity, using intents, fragments<br>**Handle Images, Listview And Menu**: ImageView, ImageSwitcher, ListView, Menu, and its types**,** Designing menu in XML, Option menu, Context menu, popup menu, **Screen** Navigation, RecyclerView, Interaction of Views<br>**Implementing Data Persistence**: Data Storage-Shared Preference, Internal And External Storage Storing Data Using SQLite Databases, Content Provider, Firebase Real-Time Data | **15** |

**References:**

- Alex Forrester,How to Build Android Apps with Kotlin: A hands-on guide to developing, testing, and publishing your first apps with Android, Packt Publishing, 2021
- Android Programming: Crafting UI/UX using Kotlin, SYBGENLearning,

**Additional References:**

- Dawn Griffiths,Head First Android Development: A Learner's Guide to Building Android Apps with Kotlin 3rd Edition, O'Reilly Media, 2021

| |
|---|
| ● Neil Smyth, Android Studio 4.2 Development Essentials - Kotlin Edition: Developing Android Apps Using Android Studio 4.2, Kotlin and Android Jetpack, Payload Media, 2021<br>● John Horton,Android Programming with Kotlin for Beginners, Packt Publishing, 2019<br>● Marcin Moskala,Android Development with Kotlin: Enhance your skills for Android development using Kotlin, Packt Publishing |
| |

| Course Code | SEC SEM-IV Practical | Credits | Lectures/ Week |
|---|---|---|---|
| 24CSSC451 | Android Application Development Practical | 2 | 4 |

**Course Outcomes:**

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

CO1:Remember the basic concepts of Android app development, including activities, fragments, intents, and layouts.

CO2: Understand both the basic and advanced concepts of Kotlin.

CO3:Applying basic UI components, including buttons, text views, edit texts, and image views.

CO4: Compare and analyze use of different concepts to solve different types of problems.

| Unit | Topics |
|---|---|
| 1 | i. Write a program using Kotlin to implement control structures and loops.<br>ii. Write a program to implement object-oriented concepts in Kotlin. |
| 2 | Create an Android application to design screens using different layouts and UI including Button, Edittext, Textview, Radio Button etc. |
| 3 | Create an Android application to demonstrate implicit and explicit intents |
| 4 | Create an application to create Image Flipper and Image Gallery. On click on the image displays the information about the image. |
| 5 | i. Create an Android application to demonstrate the use of Broadcast listeners.<br>ii. Create an Android application to create and use services. |
| 6 | Create an Android application to demonstrate XML based animation |
| 7 | Create a media player application in android that plays audio. Implement play, pause, and loop features. |
| 8 | Create an Android application to demonstrate the different types of menus.<br>a. Pop-up Menu b. Context Menu c. Option Menu |

| | |
|---|---|
| **9** | Create a suitable Android application to store and retrieve data in the SQLite database. |
| **10** | Create a suitable Android application to work with Firebase for storing and manipulating data. |

| Course Code | CEP | Credits | HOURS |
|---|---|---|---|
| **24CSCEP4** | **Foundations of Community Engagement** | **2** | **60** |

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**<br>- Overview of theories and models<br>- Importance of interdisciplinary approaches |
| Social Issues Analysis | - Guest lecture by a social scientist or experts from diverse sectors<br>- Group discussion and analysis of contemporary social issues |
| Community Needs Assessment | -Theory on needs assessment methodologies<br>- Field visit for practical application |
| Stakeholder Engagement | - Guest lecture from a community organizer<br>- 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<br>- Timeline for implementation |
| Field Work Skills Training | - Training in data collection, interviewing, and observation<br>- Practical exercises in the community |
| Ethical Considerations in Community Engagement | - Guest lecture on ethical dilemmas in community work<br>- 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 = 1 Hour**

**For field engagement/ Field Project: 1 Credit = 2**

**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.

**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:**<br>***(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** |
| **Q.1 B)** | | **05** |
| **Q.1 C)** | | **03** |
| **Q.1 D)** | | **02** |

| | | |
|---|---|---|
| **OR** | | |
| **Q.1 P)** | | **05** |
| **Q.1 Q)** | | **05** |
| **Q.1 R)** | | **03** |
| **Q.1 S)** | | **02** |
| | | |
| **Q.2 A)** | | **05** |
| **Q.2 B)** | | **05** |
| **Q.2 C)** | | **03** |
| **Q.2 D)** | | **02** |
| **OR** | | |
| **Q.2 P)** | | **05** |
| **Q.2 Q)** | | **05** |
| **Q.2 R)** | | **03** |
| **Q.2 S)** | | **02** |

**III. Practical Examination**

- Each core subject carries 50 Marks. (20 Marks internal +30 Marks external:- 20 Marks question, 5 journal +5 marks viva)
- 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:**

1. **To pass the examination, attendance is compulsory in both Internal & External (Theory + Practical) Examinations.**
2. **There is separate passing in internal and external theory courses.**