Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

NEP Compliant Structure & Syllabus of

Department of Computer Science and Engineering (Data Science)

Pattern 'A-25'

S.Y. B.Tech.

Effective from Academic Year 2026-27

Prepared by: - Board of Studies in Computer Science and Engineering – Data Science

Approved by:-Academic Board, Vishwakarma Institute of Technology

Dr. Deepa Abin
Chairman-Bos

Dr. Parikshit Mahalle
Dean-Academics

Dr. Rajesh Jalnekar
Chairman-Academic Board

Bansilal Ramnath Agarwal Charitable Trust's
VISHWAKARMA INSTITUTE OF TECHNOLOGY–PUNE
(Autonomous Institute affiliated to Savitribai Phule Pune University)

Vision & Mission Statement

Institute

Vision:

Excellence in Technical Education with Holistic Development

Mission:

- To make the industry ready graduating engineers with high human values
- To impart technical, social, innovative, and entrepreneurial skills of the highest standards.
- To prepare passing graduates for higher studies and high quality research.

Department of CSE (Data Science)

Vision:

Excellence in Data Science to empower the future of technology with holistic development

Mission:

- To equip students for industry, academia, research, and entrepreneurship by offering excellent education in emerging data science techniques.
- To empower students with ethical values and ensure their positive impact on society and beyond.
- To envision students with inter disciplinary skill sets, fully equipped to address the evolving needs of society.

Program Outcomes

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change

Academic Information–Please visit www.vit.edu

PSOs-Program Specific Outcomes

PSO-1: Apply Data Science techniques to extract valuable insights from real-world data for informed decision-making.

PSO-2: Apply Machine Learning and AI concepts for predictive modeling and intelligent system development.

PEOs-Program Educational Objectives

PEO-1: To excel in a professional career in artificial intelligence, data science engineering and related interdisciplinary domains by applying advanced knowledge and practical skills.

PEO-2: To demonstrate a strong foundational understanding that supports higher education, research endeavors, and innovation in emerging areas of technology.

PEO-3: To embody professional ethics, a commitment to lifelong learning, leadership qualities, and a collaborative spirit aimed at contributing meaningfully to society and promoting environmental sustainability.

Nomenclature for Teaching and Examination Assessment Scheme AY [2026-27]

Sr No.	Category	Head of Teaching/Assessment	Abbreviation used
1	Teaching	Theory	Th
2	Teaching	Laboratory	Lab
3	Teaching	Tutorial	Tut
4	Teaching	Open Elective	OE
5	Teaching	Multi-Disciplinary	MD
6	Teaching	Computer Science	CS
7	Assessment	Laboratory Continuous Assessment	CA
8	Assessment	Mid Semester Assessment	MSA
9	Assessment	End Semester Assessment	ESE
10	Assessment	Home Assignment	HA
11	Assessment	Course Project	CP
12	Assessment	Group Discussion	GD
13	Assessment	Power Point Presentation	PPT
14	Assessment	ClassTest–1	CT1
15	Assessment	ClassTest–2	CT2
16	Assessment	Mid Semester Examination	MSE
17	Assessment	End Semester Examination	ESE
18	Assessment	Written Examination	WRT
19	Assessment	Multiple Choice Questions	MCQ
20	Assessment	Laboratory	LAB

Contents

FF No.: 653

Structure of Second Year CSE-DS for Academic Year 2026-27

Course Code	Course Name	Type	Teaching Learning Scheme				
			Th	Tut	Lab	Hrs. / Week	Credits
Module 3 - Semester 1							
DS2003	FUNDAMENTALS OF DATA STRUCTURE AND ALGORITHMS	**DC**	3	0	2	5	4
DS2004	FUNDAMENTALS OF DATA SCIENCE	**DC**	2	0	2	4	3
DS2005	OBJECT ORIENTED PROGRAMMING	**DC**	2	0	2	4	3
DS2006	COMPUTER NETWORKS	**DC**	2	0	2	4	3
MM0404	PROBABILITY AND STATISTICS	**IEL**	2	1	0	3	3
HS2002	FROM CAMPUS TO CORPORATE - 1	**IEL**	2	0	0	2	2
HS2001	REASONING AND APTITUDE DEVELOPMENT - 3	**DC**	1	0	0	1	1
DS2007	DESIGN THINKING - 1	**SM**	0	1	0	1	1
DS2008	ENGINEERING DESIGN AND INNOVATION - 1	**SM**	0	0	4	4	2

Assessment Scheme

Course Code	Course Name	ESE TH (W)	ESE (R)	CVV	CP	LAB	GD/ PPT/ HA	MSE (W)	MSE (R)	ESE	PRACT + CVV	Total
DS2003	FUNDAMENTALS OF DATA STRUCTURE AND ALGORITHMS	40	-	-	20	-	-	-	-	-	40	100
DS2004	FUNDAMENTALS OF DATA SCIENCE	40	-	20	30	10	-	-	-	-	-	100
DS2005	OBJECT ORIENTED PROGRAMMING	40	-	20	30	10	-	-	-	-	-	100
DS2006	COMPUTER NETWORKS	-	-	20	30	10	-	40	-	-	-	100
MM0404	PROBABILITY AND STATISTICS	-	-	20	20	-	20	40	-	-	-	100
HS2002	FROM CAMPUS TO CORPORATE - 1	-	50	-	-	-	-	-	50	-	-	100
HS2001	REASONING AND APTITUDE DEVELOPMENT - 1	-	-	-	-	-	-	-	-	100	-	100
DS2007	DESIGN THINKING - 1	-	-	-	-	-	-	-	-	100	-	100
DS2008	ENGINEERING DESIGN AND INNOVATION - I	-	70	-	-	-	-	-	30	-	-	100

Dr. Deepa Abin
Chairman-BoS

Dr. Parikshit Mahalle
Dean Academics

FF No.: 654-A

DS2003 : FUNDAMENTALS OF DATA STRUCTURE AND ALGORITHMS

Teaching Scheme	Examination Scheme
Credits : 4	CP: 20 Marks
Lectures : 3 Hrs/week	PRACT+CVV: 40 Marks
Practical : 2 Hrs/week	MSE (W/O): - Marks
Tutorial : -	ESE (W): 40 Marks

Prerequisites:	
●	Introduction to Programming, Discrete Mathematics, Basic Problem- Solving Skills
Course Objectives:	
●	Understand basic data structures and algorithmic complexity.
●	Implement searching, sorting, and recursion algorithms.
●	Apply advanced structures like trees, graphs, and hashing.
●	Design solutions using stacks, queues, and linked lists.
●	Compare and optimize algorithm performance.
●	Use tools to develop and test algorithmic solutions
Course Outcomes:	
	After completion of the course, student will be able to
1.	Analyze and compute time and space complexity of algorithms.
2.	Apply searching and sorting algorithms to solve computational problems.
3.	Apply stack and queue operations in applications.
4.	Develop and use various types of linked lists to manage dynamic data.
5.	Apply tree data structures for hierarchical data representation and operations.
6.	Demonstrate understanding of graph and hashing techniques and their applications

Section1:	**Topics/Contents**

Unit-1 : Introduction to Data Structures and Complexity: Abstract Data Types, need of data structures, classification of data structures, complexity analysis of algorithms using Big-O, Big-Ω and Big-Θ notations, time-space trade-off, recursion basics and applications.

(7 HRS)

Unit-2 : Arrays and Searching & Sorting Techniques: 1D and 2D arrays, sparse matrix representation, Simple and Fast Transpose, linear search, binary search, bubble sort, selection sort, insertion sort, merge sort, quick sort, time complexity of all techniques.

(7 HRS)

Unit-3 : Stacks and Queues: Stack operations and applications like expression evaluation, infix to postfix conversion, prefix conversion, implicit and explicit stack, recursion using stack; queue operations, circular queue, priority queue, deque, applications of queues in real-world.problems. **(7 HRS)**

Section2:	Topics/Contents

Unit-4 : Linked Lists: Singly linked list, doubly linked list, circular linked list, operations (insert, delete, traverse, search), applications such as polynomial operations, memory management using dynamic allocation. **(7 HRS)**

Unit-5 : Trees: Tree terminologies, binary trees, binary search trees (BST), tree traversals (inorder, preorder, postorder), AVL trees, expression trees, heap trees, applications in decision making and file systems. **(7 HRS)**

Unit-6 : Graphs and Hashing: Graph representation using adjacency list and matrix, BFS, DFS, applications of graphs in networking and social media, hash tables, collision resolution techniques: chaining and open addressing. **(7 HRS)**

List of Practical's: (Any 10)

1. Design a recursive program to compute the factorial of a given number, illustrating how function calls build up and unwind like a stack.
2. Simulate bed allocation in a hospital ward using a sparse matrix where only a few beds are occupied, and display the occupied vs. available status efficiently.
3. Develop a phonebook application where users can search for contacts using both linear and binary search methods. Compare the search performance for small vs. large contact lists.
4. Build a shopping cart system where products can be sorted by price or ratings using various sorting algorithms. Measure and compare the time efficiency of each method for large orders.
5. Create a calculator that validates and evaluates arithmetic expressions using stacks. Implement infix to postfix conversion and postfix evaluation to simulate real-time computation.
6. Design a queue-based ticketing system (like in banks or hospitals) where customers are issued tokens and served in their order of arrival. Use a circular queue to manage overflow efficiently.
7. Simulate how train coaches (bogies) are connected using different linked lists. Each coach is a node; use singly, doubly, and circular lists to represent and manipulate various train configurations such as adding, removing, and navigating between coaches.
8. Create a contact management system where names are stored in a Binary Search Tree (BST). Implement insertion, deletion, and search functionalities to maintain a

well- organized directory.

9. Design a city navigation simulator using graphs where roads between landmarks are represented as edges. Use BFS and DFS to find paths from one place (e.g., home) to another (e.g., hospital or restaurant).
10. Implement a phone directory system using hash tables where each contact is stored by name. Handle collisions using separate chaining and provide options to add, delete, and search for contacts efficiently.
11. Develop a system to manage a library shelf where book titles are stored and sorted alphabetically. Use sorting techniques like insertion and quick sort and compare their performance.
12. Simulate an undo-redo feature of a text editor using two stacks. Enable users to type, undo recent changes, and redo actions as seen in common writing tools.
13. Create a task scheduling system where each task has a priority level. Use a priority queue to ensure higher priority tasks (e.g., emergency alerts) are served before others.
14. Simulate a memory management module where memory blocks are allocated and freed using linked lists. Implement dynamic memory allocation.
15. Build a laundry basket simulator where clothes are added one by one and removed in reverse order using a stack. Display the order in which clothes are removed for washing.
16. Develop an inventory management system for a small retail store where products are stored in a hash table using their IDs as keys. Implement features to add, search, update, and remove products efficiently while resolving collisions using open addressing.

List of Projects:

1. Smart Hospital Management System (Description: Manage patient queues, emergency prioritization, bed allocation using sparse matrix, and patient search with hashing.)

2. E-Commerce Cart and Inventory System (Description: Handle a shopping cart where products are sorted by price/rating, inventory tracked, and undo-redo implemented for cart updates.)

3. City Route Planner and Navigation Assistant (Description: Simulate city paths with graphs and implement BFS/DFS to find routes between locations.)

4. Digital Library Book Manager (Description: Store and manage book records,

search by title, sort alphabetically, and track borrow/return actions.)

5. Student Academic Dashboard (Description: Maintain student marks, attendance, course enrollments, and performance summaries.)

6. Login and Credential Manager (Description: Manage usernames and passwords with hashing. Include features like login attempts tracking and user lookup.)

7. Supermarket Billing System (Description: Simulate billing with product scanning, subtotal calculation, discounts, and inventory updates.)

8. To-Do List Manager (Description: Let users add, remove, and track tasks. Include priority tasks and optional reminders or undo functionality.)

9. Expression Solver and Visualizer (Description: Create a calculator that converts infix expressions to postfix and evaluates them step-by-step.)

10. Railway Coach Management Simulator (Description: Model how coaches are connected in a train using linked lists. Allow coach addition, removal, and navigation.)

Text Books:

1. E. Horowitz, S. Sahni, D. Mehta; Fundamentals of Data Structures in C, 2nd Edition, Universities Press, 2017.
2. Y. Langsam, M. Augenstein, A. Tannenbaum; Data Structures Using C and C++, 2nd Edition, Pearson Education, 2006.

Reference Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein; Introduction to Algorithms, 4th Edition, MIT Press, 2022. https://mitpress.mit.edu/9780262033848/
2. Brad Miller, David Ranum; Problem Solving with Algorithms and Data Structures using Python; Open Book Project; 3rd Edition, 2023. https://runestone.academy/ns/books/published/pythonds/index.html
3. Narasimha Karumanch, Data Structures and Algorithms Made Easy, Publisher CareerMonk Publications, 2024

Moocs Links and additional reading material:

1. Prof. Naveen Garg; Data Structures and Algorithms; NPTEL – IIT Delhi; https://nptel.ac.in/courses/106/102/106102064.
2. UC San Diego, National Research University Higher School of Economics; Data Structures; Coursera; https://www.coursera.org/learn/data-structures

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2
CO1	3	3	1	1	2	1	-	-	-	-	1	3	-
CO2	3	3	2	2	2	-	-	-	-	-	1	3	-
CO3	3	2	3	2	3	-	-	1	1	-	1	3	-
CO4	3	3	3	2	3	-	-	1	1	-	1	3	-
CO5	3	3	3	2	3	1	-	-	-	-	1	2	2
CO6	3	3	2	2	3	2	-	1	1	-	1	2	3
Avg	**3.0**	**2.8**	**2.3**	**1.8**	**2.7**	**0.7**	**0.0**	**0.5**	**0.5**	**0.0**	**1.0**	**2.7**	**1.2**

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓	✓	-	-	-	-	-	-
CO2	-	✓	✓	✓	-	✓	-	-	-
CO3	-	-	✓	✓	-	✓	-	-	-
CO4	-	-	✓	✓	-	✓	-	-	-
CO5	-	-	✓	✓	✓	✓	-	-	-
CO6	-	✓	✓	✓	✓	✓	✓	-	✓

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓	✓	✓	-	-	-	-
CO2	✓	✓	✓	✓	-	-	-
CO3	✓	✓	✓	✓	✓	-	-
CO4	✓	✓	✓	✓	✓	✓	-
CO5	✓	✓	✓	✓	✓	✓	✓
CO6	✓	✓	✓	✓	✓	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	-	-	-	-
CO2	✓	✓	-	-	-
CO3	✓	✓	✓	-	-
CO4	✓	✓	✓	✓	-
CO5	✓	✓	✓	✓	✓
CO6	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	**4, 9**	Understanding algorithm complexity and fundamental data structures strengthens quality education and builds the foundation for innovation in computing.
CO2	**4, 9**	Efficient searching and sorting algorithms improve computational efficiency and support innovation in software and digital infrastructure.
CO3	**9**	Stack and queue applications are fundamental in compiler design, operating systems, networking, and other modern computing technologies.
CO4	**9, 12**	Linked-list based dynamic memory management promotes efficient utilization of computing resources and sustainable software design.
CO5	**9**	Tree data structures enable efficient database indexing, file systems, decision-making, and intelligent computing applications.
CO6	**9, 11**	Graphs and hashing support smart transportation, communication networks, cybersecurity, and scalable digital infrastructure contributing to sustainable and smart communities.

FF No.: 654-A

DS2004: FUNDAMENTALS OF DATA SCIENCE

Teaching Scheme	Examination Scheme
Credits : 3	CP: 30 Marks
Lectures : 2 Hrs/week	CVV: 20 Marks
Practical : 2 Hrs/week	LAB: 10 Marks
Tutorial : - Hrs/week	ESE (W): 40 Marks

Prerequisites:	
●	Python programming, fundamental concepts in statistics, basic problem- solving skills
Course Objectives:	
●	Understand the evolution, scope, lifecycle, and real-world applications of data science
●	Learn data collection methods, preprocessing techniques, and feature extraction from various data sources
●	Perform exploratory data analysis and create effective data visualizations using Python and visualization tools
●	Develop basic machine learning models for regression and classification problems
Course Outcomes:	**[Publication of Paper or Patent]**
	After completion of the course, student will be able to
1.	Explore the fundamentals, applications, lifecycle, and ethical aspects of data science in various real-world domains.
2.	Apply data pre-processing and feature extraction techniques to prepare data for analysis.
3.	Analyse data using statistical summaries and visualize relationships and patterns through suitable graphical representations.
4.	Build and evaluate predictive models using appropriate techniques for classification and regression.

Section 1:	**Topics/Contents**

Unit-1: Introduction to Data Science

Overview, Evolution and Scope, Components: Data, Algorithms, Insights, Data Science Life Cycle: Data acquisition, Analysis, Action, Applications of Data Science: Healthcare, Finance, Retail & E-commerce, Manufacturing, Smart Cities & IoT & Social Media & Marketing, Case Study: Predictive Maintenance in Manufacturing – to identify the type of data requirement and to sketch a basic DS workflow. **(6 HRS)**

Unit-2: Data Collection, Cleaning, and Feature Extraction

Data Sources and Types ,Types of data: structured, unstructured, semi-structured, Data formats: numerical, categorical, time-series, text. Sources: databases, APIs, IoT sensors, web, surveys. Data Collection Techniques: Manual vs automated collection methods, Web scraping requests, Data Cleaning and Preprocessing: Handling missing values, Removing duplicates and irrelevant data, Standardizing and converting data types, Detecting and treating outliers (Z-score, IQR – theory only), Encoding categorical variables: label encoding, one-hot encoding, Feature Extraction: Understanding features, Feature selection, Feature engineering,

Converting non-numeric to numeric data for models.

Case Study: Social Media Sentiment Analysis- to process and extract features from user-generated text data to support sentiment classification. **(8 HRS)**

Section 2:	**Topics/Contents**

Unit-3: Exploratory Data Analysis using Python

Exploratory Data Analysis (EDA) Concepts, Importance of EDA in Data Science projects, Basic statistics: Mean, Median, Mode, Standard Deviation; understanding distributions: Normal, Skewed, Correlation analysis (Pearson, heatmaps).Visualization Tools and Techniques: Line plots, bar charts, histograms, scatter plots, funnel chart, heatmap, waterfall chart; matplotlib, tableau, seaborn, Plotly, D3.js, Power BI, Qlik Sense.

Emerging AI tools: Datawrapper, Flourish, Gafrana.

Case Study: Indian rainfall data (2010 to 2020)-to analyze trends across different regions.

(8 HRS)

Unit-4: Model Building and Evaluation

Model Building: Introduction to model training workflow methods. Building simple models: Linear Regression, K-Nearest Neighbors (KNN), K-means clustering. Model Evaluation Techniques Metrics for Regression: RMSE, MAE, R^2 Score. Metrics for Classification: Accuracy, Precision, Recall, F1-score, Confusion Matrix, Cross-validation.

Case Study: Student Performance Prediction- machine learning model that predicts whether a student is likely to pass or fail based on personal, academic, and family-related data. **(6 HRS)**

Text Books:

1	C. O'Neil, R. Schutt; Doing Data Science; 1st Edition; O'Reilly Media; 2013
2	J. Grus; Data Science from Scratch: First Principles with Python; 2nd Edition; O'Reilly Media; 2019
3	W. McKinney; *Python for Data Analysis*; 2nd Edition; O'Reilly Media; 2018

Reference Books:

1	C. O'Neil, R. Schutt; Doing Data Science; 1st Edition; O'Reilly Media; 2013. Accessed: Jun. 17, 2025. [Online]. Available: https://www.oreilly.com/library/view/doing-data-science/9781449363871/
2	J. Grus; Data Science from Scratch: First Principles with Python; 2nd Edition; O'Reilly Media; 2019. Accessed: Jun. 17, 2025. [Online]. Available: https://www.oreilly.com/library/view/data-science-from/9781492041122/
3	W. McKinney; Python for Data Analysis; 2nd Edition; O'Reilly Media; 2018. Accessed: Jun. 17, 2025. [Online]. Available: https://www.oreilly.com/library/view/python-for-data/9781491957653/
4	C. Müller, S. Guido; Introduction to Machine Learning with Python; 1st Edition; O'Reilly Media; 2016. Accessed: Jun. 17, 2025. [Online]. Available: https://www.oreilly.com/library/view/introduction-to-machine/9781449369880/

List of Practical's:

1. Explore a sample dataset and summarize its structure- including column names, data types, potential use cases; data cleaning and pre-processing, and analyse the distribution and central tendencies of variables and detect any skewness or

irregularities.

2. A multi-specialty hospital wants to improve patient care and reduce treatment delays. Currently, they rely on manual records and experience difficulty in identifying disease trends and patient risk factors. Analyze how Data Science can transform the hospital system by:
    - Explaining the evolution of Data Science in healthcare
    - Identifying its scope and benefits
    - Highlighting real-world applications such as disease prediction and patient monitoring
3. An online retail company is facing inconsistent sales performance across regions and seasons. Management wants to understand the entire data flow from raw data to actionable insights. Design a complete Data Science lifecycle for analyzing sales data:
    - Define how data is collected (transactions, user behavior)
    - Describe cleaning, EDA, modeling, and deployment stages
    - Suggest how insights can improve business decisions
4. A smart city project collects data from multiple sources like traffic sensors, weather APIs, and public transport systems. However, the data is scattered and inconsistent. Develop a system to:
    - Collect data from at least two sources (CSV + API)
    - Integrate and merge datasets
    - Prepare a unified dataset for analysis
5. A banking system has transaction data with missing values, duplicate entries, and inconsistent formats. These issues are affecting fraud detection models. Perform data preprocessing by:
    - Handling missing values appropriately
    - Removing duplicate and inconsistent records
    - Detecting and treating outliers
    - Applying normalization and encoding techniques
6. A retail company wants to understand customer buying behavior but the existing dataset lacks meaningful features for analysis. Enhance the dataset by:
    - Creating new features (e.g., total spending, frequency of purchase)
    - Performing feature scaling and selection
    - Identifying the most relevant features for prediction
7. A university wants to analyze student performance to identify factors affecting academic success. Perform EDA to:
    - Analyze distribution of marks
    - Identify correlations between subjects
    - Detect outliers and unusual patterns
8. A company's management team needs a dashboard to quickly understand sales trends, profits, and regional performance. Design visualizations that:
    - Represent sales trends over time

- Compare region-wise performance
- Highlight key insights using charts and graphs

9. A real estate company wants to predict house prices based on features like location, size, and number of rooms. Build a regression model to:
   - Train on historical housing data
   - Predict prices for new houses
   - Evaluate performance using error metrics

10. A bank wants to automate loan approval decisions based on applicant details such as income, credit score, and employment status. Develop a classification model to:
    - Predict whether a loan should be approved or rejected
    - Evaluate using accuracy and confusion matrix

11. A startup wants to leverage data to improve its decision-making but lacks a structured analytics pipeline. Perform a complete Data Science workflow:
    - Data collection
    - Data cleaning and preprocessing
    - EDA and visualization
    - Model building (regression or classification)

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	3		2	2						2	1	1
CO2	3	3	2		2						1	2	1
CO3	3	2	3		3							1	2
CO4	3	2	3		3							2	1
Avg	3.0	2.5	2.0	0.5	2.5	0.0	0.0	0.0	0.0	0.0	0.75	1.5	1.25

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓	✓	✓					
CO2	✓	✓	✓	✓	✓	✓			
CO3	✓	✓	✓	✓		✓		✓	
CO4	✓	✓	✓	✓	✓	✓		✓	

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓		✓	✓			
CO2	✓	✓	✓		✓		
CO3	✓	✓	✓	✓		✓	
CO4	✓	✓	✓	✓	✓	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓		✓		✓
CO2	✓	✓	✓		✓
CO3	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4, 9	Builds a strong foundation in data science concepts (SDG 4) and promotes innovation through understanding real-world applications (SDG 9).
CO2	9	Data collection, preprocessing, and feature engineering are fundamental to intelligent data-driven systems and industrial innovation (SDG 9).
CO3	4, 9	Exploratory data analysis and visualization enhance analytical skills (SDG 4) and support data-driven innovation and decision-making (SDG 9).
CO4	8, 9	Machine learning models enable intelligent automation, predictive analytics, and industrial applications, contributing to innovation (SDG 9) and economic growth through AI-driven solutions (SDG 8).

FF No.: 654-A

DS2005: OBJECT ORIENTED PROGRAMMING

Teaching Scheme	Examination Scheme
Credits : 3	CP: 30 Marks
Lectures : 2 Hrs/week	LAB: 10 Marks
Practical : 2 Hrs/week	CVV: 20 Marks
Tutorial : - Hrs/week	ESE (W): 40 Marks

Prerequisites:	
•	Introduction to Programming in C language, Basic Problem-Solving Skills
Course Objectives:	
•	To explain the fundamental concepts of object-oriented programming and apply them to develop modular programs using classes and objects in C++.
•	To implement and analyze object-oriented features such as constructors, destructors, and operator overloading for developing reusable and efficient components.
•	To examine and evaluate the use of inheritance, polymorphism, and virtual functions in designing extensible and maintainable software systems.
•	To apply exception handling and file input/output mechanisms to design and develop robust and persistent C++ applications
Course Outcomes:	
	After completion of the course, student will be able to
1.	Demonstrate understanding of object-oriented programming fundamentals and implement modular programs using classes and objects in C++.
2.	Apply constructors, destructors, operator overloading, and analyze class relationships to design solutions using inheritance and polymorphism
3.	Use templates and STL components to create generic and reusable software modules.
4.	Implement exception-handling and file I/O techniques and evaluate their effectiveness in developing robust and persistent applications

Section1: Topics/Contents

Unit-1 : Introduction to Object Oriented Programming

Fundamentals of Object-Oriented Programming (OOP) in C++: Variables, Data Types, and Operators, Control Structures, Loops and Iteration, Functions and Modular Programming, Basics of Console Input and Output Class, Dynamic Memory Allocation.

Overview of OOPs Principles, Classes & Objects, Creation & destruction of objects, Data Members, Member Functions, Access Specifier, this Pointer, Constructor & Destructor, Static class member, Friend class and functions, Function Overloading, Introduction to OperatorOverloading. **(8 HRS)**

Unit-2: Object Oriented Programming Principles

Inheritance, Types of Inheritance, Base and Derived class Constructors, Down casting and upcasting, Function overriding, Virtual functions, Polymorphism, Pure virtual functions, Virtual Base Class, C++ Class Hierarchy **(6 HRS)**

Section2:	Topics/Contents

Unit-3 : Templates, Operator Overloading and Standard Template Library

Function Templates, Class Templates, Operator Overloading, Overloading << and >> operators, Namespace usage and scope resolution, Standard Template Library (STL) components: vector, list, map, set, stack, queue. **(8 HRS)**

Unit-4:- Exception and File Handling in C++

Exception Handling: Benefits, Try and catch block, throw statement, pre-defined exceptions in C++, writing custom Exception class, Stack Unwinding, File Handling: Streams, File Input/Output, Text File Handling, Binary File Handling, Error Handling during file operations,UseCases . **(6 HRS)**

List of Practicals

1. Fundamentals of C++

a) Design an electricity billing system to calculate total cost based on unit slabs and taxes.

b) Develop a temperature monitoring module that converts Celsius readings to Fahrenheit for reporting.

c) Create a land measurement system to compute area of rectangular plots for municipal records.

2. Decision Making and Loops

a) Develop a voter eligibility system that processes multiple users and generates a report of eligible voters.

b) Create a student result analysis system that categorizes students into pass/fail/distinction using conditions and loops.

c) Simulate population growth using Fibonacci logic for a given number of years.

3. Functions and Modular Programming

a) Build a scientific calculator module implementing factorial (recursion) and prime number-checking.

b) Develop a reusable function-based system to generate Fibonacci series and analyze number patterns.

c) Create a modular application for validating user inputs (e.g., password strength or numeric constraints).

4. Classes and Objects

a) Design a geometry tool using a Rectangle class for area and perimeter calculations in construction planning.

b) Develop a product management system using classes to store and display product details.

c) Implement a student information system using class objects for storing and retrieving data.

5. OOP Concepts (Static, this pointer, Friend Function)

a) Develop a banking system using static members to track total accounts created.

b) Create an application demonstrating use of this pointer to resolve ambiguity in employee records.

c) Build a system using friend functions to access and manipulate private data of a class.

6. Operator Overloading

a) Implement arithmetic operator overloading for a Complex number or financial transaction system.

b) Overload comparison operators to compare objects like students (marks) or products (price).

c) Design a system where objects can be added or compared using intuitive operators.

7. Inheritance and Polymorphism

a) Build a shape hierarchy (Circle, Rectangle, Triangle) using inheritance and virtual functions for area calculation.

b) Extend a banking system with Savings and Current accounts using runtime polymorphism.

c) Develop an employee management system with different roles (Manager, Engineer) using inheritance.

8. Templates and STL

a) Implement a generic function template to find maximum values in datasets (marks, salaries).

b) Develop a system using STL vector and list to manage dynamic collections (e.g., student records).

c) Use map and set to create a key-value based application like employee database or dictionary.

9. Exception Handling

a) Design an ATM system that handles errors like insufficient balance and invalid inputs using try-catch.

b) Create a program with user-defined exceptions for invalid transactions or data entries.

c) Implement input validation system that prevents crashes using exception handling.

10. File Handling

a) Develop a student record system using text files for storing and retrieving data.

b) Implement a payroll system using binary files for efficient storage of employee records.

c) Add error handling for file operations such as file not found or read/write failure.

List of Projects:

1. Multi-Player Game Engine Framework (Using OOP + Polymorphism + STL)
Description:
Design a modular game engine that supports multiple games like Tic Tac Toe, Snake, and Chess. Use inheritance and polymorphism for game rules, STL containers to manage player profiles and scores, and templates for generic game logic.

2. University ERP Simulator (Object-Oriented + File Handling + Exception Handling)
Description:
Simulate a mini ERP with modules for student info, staff, timetable, fee tracking, and attendance. Use classes and virtual functions, file storage for persistent records, and exception handling for login errors, data corruption, etc.

3. Compiler Intermediate Code Generator (Advanced OOP + Stack + STL Map)
Description:
Parse simplified mathematical expressions and generate three-address code (TAC) or postfix code. Use stack, maps for symbol tables, and template functions for evaluation logic.

4. Airline Reservation & Flight Management System (Linked Lists + Operator Overloading + File Streams)
Description:
Allow seat reservation, cancellation, and route planning. Overload operators for seat comparisons, use linked lists for dynamic seat mapping, and manage historical booking data via files.

5. Blockchain-based Voting System (File Handling + Hashing + STL)
Description:
Simulate a simple blockchain to securely record votes. Use file handling to persist blocks, hash functions for verification, and vectors/queues/maps from STL to manage users, blocks, and votes.

6. Dynamic Code Snippet Organizer (Templates + File Parsing + OOP)
Description:
A tool to manage C++/Python snippets. Use template classes to generalize storage, file handling to import/export snippets, and OOP to manage languages, tags, and categories.

7. AI-Powered Chatbot Framework (Polymorphism + STL + File Handling)
Description:
Design a rules-based chatbot with polymorphic response models (default, emotion-based, logic-based), maps and vectors for pattern matching, and text file storage for chat history and knowledge base.

8. Social Media Post Scheduler (Priority Queue + Templates + Exception Handling)
Description:
Simulate scheduling of posts with time-based priority. Use STL priority queues, template-based scheduling logic, and exception handling for invalid scheduling formats.

9. Memory Management Visualizer (Pointer Arithmetic + Custom Allocators).
Description:
Create a tool that visualizes memory allocation in arrays, objects, and pointers. Implement a custom memory allocator, simulate new/delete, and show fragmentation via console visualization.

10. Trading Bot Simulator for Stock Market (Templates + STL + File I/O + Inheritance).
Description:

Design a simulated trading bot that makes buy/sell decisions. Use template-based data structures for financial instruments, inheritance for bot strategies, and files to import/export market data.

Reference Books:

1. Behrouz A. Forouzan, Richard F. Gilberg, "COMPUTER SCIENCE – A Structured Programming approach using C", Indian Edition, Thomson, 3rd edition, 2007, ISBN: 9788131517888.
2. Bjarne Stroustrup, ― The C++ Programming language‖, Third edition, Pearson Education.1997, ISBN 9780201889543. 3. Kernighan, Ritchie, "The C Programming Language", Prentice Hall of India, 1988. ISBN 0-13-110362-8.
3. Robert Lafore, ―Object-Oriented Programming in C++, fourth edition, Sams Publishing, 2002, ISBN:0672323087 (ISBN 13: 9780672323089)
4. Herbert Schildt, ―C++ The complete reference, Eighth Edition, McGraw Hill Professional, 2011, ISBN:978-00-72226805
5. E. Balagurusamy- Object-oriented programming with C++, fourth edition, Mc Hill Professional,2008, ISBN 978-0-07-066907-9

MOOC Links and additional reading material:

1. Prof. Partha Pratim Das Programming in C++, IIT Kharagpur https://onlinecourses.nptel.ac.in/noc21_cs02/preview

2. Programming in C++: A Hands-on Introduction Specialization https://www.coursera.org/specializations/hands-on-cpp

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	3	-	-	-	2	2	2	2	3
CO2	3	3	2	3	3	-	-	-	2	2	2	2	3
CO3	3	3	3	3	3	-	-	-	2	2	3	3	3
CO4	3	3	2	3	3	-	-	-	2	2	2	2	3
Avg	**3.0**	**3.0**	**2.2**	**2.7**	**3.0**	**0.0**	**0.0**	**0.0**	**2.0**	**2.0**	**2.2**	**2.2**	**3.0**

CO – WK Mapping (Knowledge & Attitude Profiles)

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓	✓	–	–	–	–	–	–
CO2	–	✓	✓	✓	–	✓	–	–	–
CO3	–	✓	✓	✓	–	✓	–	✓	–
CO4	–	✓	✓	✓	✓	✓	–	✓	✓

CO – WP Mapping (Complex Engineering Problems)

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓	✓	✓	–	–	–	–
CO2	✓	✓	✓	✓	–	–	–
CO3	✓	✓	✓	✓	✓	–	–
CO4	✓	✓	✓	✓	✓	✓	–

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	–	–	–	–
CO2	✓	✓	✓	–	–
CO3	✓	✓	✓	✓	–
CO4	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4, 9	Understanding OOP fundamentals strengthens quality education and builds a foundation for modern software development and innovation.
CO2	4, 9	OOP concepts such as inheritance and polymorphism promote reusable, scalable, and innovative software systems.
CO3	9	Templates and STL improve software efficiency, reusability, and industrial software development, supporting innovation and digital infrastructure.
CO4	9, 12	Exception handling and file management improve software reliability, efficient resource utilization, and sustainable software engineering practices.

FF No.: 654-A

DS2006: COMPUTER NETWORKS

Teaching Scheme	Examination Scheme
Credits : 3	CP: 30 Marks
Lectures : 2 Hrs/week	CVV: 20 Marks
Practical : 2 Hrs/week	MSE (W): 40 Marks
Tutorial : - Hrs/week	LAB: 10 Marks

Prerequisites:	
●	Basics of Computer organization and Digital electronics
Course Objectives:	
●	Describe basic concepts of switching, network topologies, and modules.
●	Understand different Transmission medias and differentiate Guided and Unguided medias.
●	Apply the concept of Data Link and Network layer to simulate and analyze different networks.
●	Differentiate the role of Transport and Application layer in OSI and TCP/IP network models.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Discuss the fundamental concepts of networks, network topologies, and modules.
2.	Understand the different Transmission medias and basics of physical layer protocols.
3.	Use the concept of Data Link and Network layer to simulate and analyze different networks.
4.	Differentiate the role of Transport and Application layer in OSI and TCP/IP network models

Section1:	**Topics/Contents**

Unit-1:

Networking Fundamentals and Addressing Techniques Basics of Computer Networks: Switching (Circuit & Packet), Network Topologies architecture and components, Network types, OSI/TCP-IP models, Introduction to Subnets, IP Addressing and its types (in detail Ex. types of addressing for classful and classless subnets), Basic Protocol stack.

Unit-2:

Physical Layer Transmission media: Guided and Unguided Standard terminologies. Introduction to the Physical Layer and its role in the OSI model, Protocols. **(16 HRS)**

Section2:	**Topics/Contents**

Unit-3:

Data Link and Network Layer Protocols Data Link Layer protocols, MAC addressing, Error detection and correction techniques, Hop to-hop delivery and framing, Routing (Static and Dynamic), Router configuration basics, Network Layer Protocols: IPv4 and IPv6. Subnetting (Classful/Classless)

Unit-4:

Transport and Application Layers Transport Layer responsibilities, TCP vs UDP, Socket Programming basics using TCP/IP, Application Layer protocols and services, Session and Presentation layer concepts, Overview of well-known application protocols (HTTP, FTP,

DNS, SMTP), DHCP and DNS server configuration using simulation tools. Case Study: Data Centre Network Architecture. **(12 HRS)**	
Text Books:	
1	Atul Kahate, "Cryptography and Network Security", McGraw Hill Publication, 2nd Edition, 2008, ISBN : 978-0-07-064823-4.
2	Stallings William., "Data and Computer Communications", Sixth Edition, Prentice Hall of India, 2014, 10th edition, ISBN 978-0-133-50648-8
Reference Books:	
1	Andrew S. Tanenbaum ,"Computer Networks", Pearson, 1994, ISBN-13: 978-0-13 212695-3
2	Fourouzan B., "Data Communications and Networking", 5th edition, McGraw- Hill Publications, Fifth Edition - 1 July 2017. ISBN-13: 978-1259064753 ISBN-10: 1259064751.

List of Practical's: (Any 6)

1. Basic LAN Design Using Star and Bus Topologies
   - Design simple LANs using switches and hubs.
   - Observe device interconnection and test communication.
2. Simulating OSI Layer Communication with Ping and Traceroute
   - Use ping and tracert to visualize data flow through OSI/TCP-IP layers.
   - Understand encapsulation and address resolution.
3. Static Routing Between Two Networks
   - Configure routers with static routes to enable communication between different subnets.
   - Test with ping and verify routing table entries.
4. Class-full Subnetting and IP Address Allocation
   - Use class-full addressing to efficiently divide a network.
   - Assign IPs to devices and verify connectivity.
5. Classless Subnetting and IP Address Allocation
   - Use VLSM/CIDR to efficiently divide a network.
   - Assign IPs to devices and verify connectivity.
6. DNS/ DHCP Server Configuration
   - Set up DNS and DHCP servers in a network.
   - Verify dynamic IP assignment and name resolution.
7. ACL-Based Packet Filtering and Access Control
   - Apply standard ACLs to permit or deny specific traffic (e.g., ICMP, Telnet) between subnets.
   - Observe restricted and allowed paths.
8. Firewall Simulation Using Extended ACLs
   - Use extended ACLs to simulate firewall behavior (e.g., block HTTP, allow DNS/FTP).
   - Apply on router interfaces to control traffic.

List of Project areas:

1. Real- time Network simulation
2. Implementation of Network protocols
3. Design and develop Network simulation tool

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO		
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2	PSO 3
CO1	2	-	-	-	-	-	-	-	-	-	-	-	-	-
CO2	2	-	-	-	-	-	-	-	-	-	-	-	-	-
CO3	2	2	2	-	2	-	-	-	1	-	-	-	-	-
CO4	2	-	-	-	-	-	-	-	-	-	-	-	-	-
Avg	2	0.5	0.5	-	0.5	-	-	-	-	-	-	-	-	-

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓								
CO2	✓								
CO3	✓	✓	✓	✓	✓	✓			
CO4	✓								

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1							
CO2							
CO3							
CO4							

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1					
CO2					
CO3	✓				
CO4					

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4	Builds the analytical foundation (SDG 4) that drives technical skills in computer Networks
CO2	4	Basic concepts (SDG 4) that needs to gain technical skills in computer Networks
CO3	4	Develop analytical and design skills in computer Networks
CO4	4	

FF No.: 654-A

MM0404: PROBABILITY & STATISTICS

Teaching Scheme	Examination Scheme
Credits : 3	CP: 20 Marks
Lectures : 2 Hrs/week	HA: 20 Marks
Practical : - Hrs/week	MSE (W): 40 Marks
Tutorial : 1 Hrs/week	CVV: 20 Marks

Prerequisites:	
●	Course Prerequisites: Basic algebra, Calculus, Logical reasoning and basic problem-solving skills
Course Objectives:	
●	To introduce students to basic statistical concepts and probability theory.
●	To enable students to model real-world uncertainty using probability distributions.
●	To train students in inferential techniques such as estimation and hypothesis testing.
●	To build foundational skills in analyzing relationships between variables using correlation and regression
Course Outcomes:	
	After completion of the course, students will be able to
1.	Explain descriptive statistics, probability concepts, and random variables.
2.	Apply discrete and continuous probability distributions to engineering problems.
3.	Perform statistical inference using estimation and hypothesis testing.
4.	Analyze and interpret relationships in data using correlation and regression.

Section1:	**Topics/Contents**
Unit 1: Descriptive Statistics & Fundamentals of Probability: Types of data: qualitative, quantitative, discrete, continuous, Frequency distribution; visualizations (histogram, bar chart, box plot), Measures of central tendency: mean, median, mode, Measures of dispersion: variance, standard deviation, coefficient of variation, Basic probability rules: axioms, complement rule, addition rule, Conditional probability & independence, Bayes' theorem (simple engineering examples) **(7 HRS)** **Unit 2: Random Variables & Common Distributions**: Random variables: discrete & continuous, Probability Mass Function (PMF), Probability Density Function (PDF), Cumulative Distribution Function (CDF), Expectation, variance, and properties, Discrete distributions: Bernoulli, Binomial, Poisson, Continuous distributions: Uniform and Exponential, Applications in engineering **(7 HRS)**	
Section2:	**Topics/Contents**
Unit 3: Normal Distribution & Sampling Theory: Normal distribution: standard normal, Z-scores, Applications of normal distribution (tolerances, quality control), Normal approximation to binomial & Poisson, Sampling: random sampling, sample mean, sample variance, Sampling distribution of sample mean, Standard error, Central Limit Theorem (CLT) – intuitive explanation with examples **(7 HRS)**	

Unit 4: Estimation, Hypothesis Testing, Correlation & Regression: Point estimation: properties of good estimators, Interval estimation: confidence interval for mean (σ known/unknown), Hypothesis testing: Null & alternative hypothesis, Type I & II errors, One-sample Z-test and t-test, Chi-Square (χ^2) Tests: Chi-square distribution, Test for goodness of fit, Test for independence of attributes, Applications in quality inspection and survey data, Analysis of Variance (ANOVA): Concept and assumptions of ANOVA, One-way ANOVA, F-test and interpretation, Engineering and experimental applications

(7 HRS)

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)												PSO		
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2			1					1		2	1	1	
CO2	3	3	1	1	1					1		2	2	2	
CO3	3	3	1	2	1			1		1		3	2	2	
CO4	3	3	2	2	2	1		1	1	2		3	3	3	
CO5	3	2			1					1		2	1	1	
CO6	3	3	1	1	1					1		2	2	2	
Avg	3.0	2.8	1.3	1.7	1.3	1.0		1.0	1.0	1.3		2.5	2.0	2.0	

CO – WK Mapping (Knowledge & Attitude Profiles)

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓							
CO2		✓	✓	✓					
CO3		✓	✓	✓	✓				
CO4			✓	✓	✓	✓	✓		

CO – WP Mapping (Complex Engineering Problems)

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓		✓	✓			
CO2	✓	✓	✓	✓			✓
CO3	✓	✓	✓	✓	✓		✓
CO4	✓	✓	✓	✓	✓	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓				✓
CO2	✓	✓	✓		✓
CO3	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals)

CO	SDG(s) Addressed	Justification / Linkage
CO1	4, 9	Understanding descriptive statistics, probability, and random variables strengthens quantitative reasoning, supports quality education, and provides the mathematical foundation for innovation and engineering applications.
CO2	4, 9	Applying discrete and continuous probability distributions enables students to solve engineering problems using analytical methods, fostering innovation and technological advancement.
CO3	4, 9, 16	Statistical inference through estimation and hypothesis testing promotes evidence-based decision-making, responsible research practices, and reliable engineering solutions.
CO4	4, 9, 17	Correlation and regression analysis facilitate data-driven decision-making, predictive modeling, interdisciplinary collaboration, and knowledge sharing across engineering domains.

FF No.: 654-A

HS2002: FROM CAMPUS TO CORPORATE (FCTC-1)

Teaching Scheme	Examination Scheme
Credits : 02	MSE : 50 Marks
Lectures : 2 Hrs/week	ESE : 50 Marks

Course Objectives:	
•	Understand the organizational structure, business domains, financial concepts, and functioning of companies in the corporate environment.
•	Develop knowledge of product management, user experience, quality assurance, customer success, security, and emerging technologies used in modern industries.
•	Apply analytical thinking, problem-solving, and decision-making techniques to address industry-oriented challenges through case studies and practical scenarios.
•	Enhance professional competencies including communication, presentation, teamwork, leadership, and interpersonal skills required in multidisciplinary work environments.
•	Develop project management, planning, prioritization, and execution skills for effective management of professional tasks and projects.
•	Cultivate professional ethics, adaptability, critical thinking, and lifelong learning to achieve continuous personal and professional growth.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Explain the organizational structure, business processes, financial concepts, and product development lifecycle in the corporate ecosystem.
2.	Analyse workplace problems and recommend suitable solutions using concepts of product management, quality, customer success, security, and emerging technologies.
3.	Apply critical thinking, problem-solving techniques, AI awareness, and project management principles to solve industry-oriented case studies.
4.	Demonstrate effective communication, teamwork, leadership, and presentation skills while working in professional and multidisciplinary environments.
5.	Apply project planning, prioritization, and management techniques for successful execution of professional assignments and collaborative projects.
6.	Demonstrate professional ethics, social responsibility, adaptability, and lifelong learning necessary for continuous professional development in a dynamic corporate environment.

Vertical 1: COMPANIES & INDUSTRY

Topic 1: Understanding Companies

- Understand how to analyse a company-based domains, customers, offering and strategy
- Understand how to use this knowledge to contribute earlier and faster.

Topic 2: Understanding Company Financials (and AOP Budgeting)

- Understand components of company P&L and Balance Sheet
- Be able to assess financial health of a company
- How do companies build AOP. Top down and bottom up

- Understand the funding avenues

Vertical 2: Application Of Knowledge

Topic 1: Product Management

- Understand product roadmap creation
- Understand product growth strategies
- Understand product discovery techniques
- Understand backlog prioritization techniques
- Be able to write requirements in detail

Topic 2: Understanding UX And UI

- Understand what usability, user interaction design and consumer journeys mean
- Compare and contrast UX with user interface. Use several industry examples
- technique of building a good UX

Topic 3: Releasing A Product To The Market

- Understand what it takes to release a product to the market by product and engineering teams
- Understand what it takes for a product to be successful with the customer from external go-to market perspective.

Topic 4: Quality – Understanding Beyond The Theory

- Understand how to look at product quality
- Understand how to approach building a good quality product
- Understanding Testing and Test Automation
- Build a Test Plan
- Career options in Quality

Topic 5: Customer Success Management

- Concept of Customer Success
- What is CSAT , NPS , RATING, REVIEWs and how should one track Customer Value
- Understand what issues mean in the industry
- Understand how to identify and prioritize issues and ways to resolve them.

Topic 6: Security And Privacy

- Understand Security concepts (Threat, Vulnerability, Risk)
- Understand about Threat Modelling
- Understand about Security Engineering Approach
- Be able to prepare security test plan

Topic 7: Artificial Intelligence (AI) – The Evolution

- Introduction to AI. AI is pervasive today. Origins of AI and ML Potential and possibilities
- Machine Learning. How is it different from traditional programs? Extensive practical applications over the last decade or so
- Deep Learning. Computer Vision and Natural Language Processing

Vertical 3: PROFESSIONAL SKILLS

Topic 1: Curiosity And Inquisitiveness

- What is Curiosity and Understand the importance of asking questions
- Review Industry Examples
- Understand techniques of asking thought provoking questions

- Tips and tricks to ask effective questions

Topic 2: Articulation And Communication

- Understand different types of communications
- Understand barriers to communications
- Understand why articulation is important
- How do you make Impactful Introductions

Topic 3: Problem Breakdown and Resolution Using Mind Maps

- What are mind maps
- Tips on creating Mind Maps
- Real-life case of Problem Breakdown
- Prioritizing a long list. Concept and Technique
- Live workshop on understanding constraints and prioritizing and long to-do list

Topic 4: Effective Project And Program Management

- Understand the key concepts within project management
- Understand key aspects of planning
- Understand the link between planning and execution
- Case study based Proj and Prog Management hands on workshop

Topic 5: Developing Critical Thinking Using Mental Models

- Why Critical Thinking
- What are the barriers to Critical Thinking
- Technique and Steps in Critical Thinking
- Why Mental Models
- Using Mental Models for sharper decisions

CO-PO Mapping

CO	Program Outcomes (PO)											PSO		
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO 2	PSO 3
CO1	2	2	–	–	1	–	–	–	1	–	1			
CO2	2	3	2	1	2	2	1	1	–	–	1			
CO3	1	3	2	2	2	1	–	2	1	2	2			
CO4	–	1	–	–	–	–	2	3	3	2	1			
CO5	–	1	2	1	1	1	3	2	2	3	2			
CO6	–	–	–	–	1	2	3	2	2	1	3			
Average														

CO – WK Mapping (Knowledge & Attitude Profiles) Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓	-	-	-	-	-	-	-
CO2	-	✓	-	✓	-	✓	-	-	-
CO3	-	✓	-	✓	-	-	-	✓	-
CO4	-	-	-	-	-	-	-	-	✓
CO5	-	-	-	-	-	-	-	✓	✓
CO6	-	-	-	-	-	-	✓	✓	✓

CO – WP Mapping (Complex Engineering Problems) Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	-	✓	-	-	-	-	✓
CO2	-	✓	-	✓	-	✓	-
CO3	-	-	✓	✓	-	-	-
CO4	-	-	-	-	-	-	-
CO5	-	✓	-	-	-	-	✓
CO6	-	✓	-	-	-	-	-

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	-	-	-	-
CO2	-	-	-	-	-
CO3	-	✓	✓	-	-
CO4	✓	✓	-	-	✓
CO5	✓	✓	-	-	-
CO6	-	-	-	✓	✓

CO – SDG Mapping (Sustainable Development Goals) Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4, 8, 9	Builds foundational understanding of organizational, financial, and product-lifecycle knowledge that prepares students for productive employment and industry participation.
CO2	8, 9	Analyzing workplace problems via product management, quality, security, and emerging technology supports innovation-driven, industry-relevant solutions.
CO3	4, 9	Develops critical thinking, problem-solving, and AI awareness through case studies — building future-ready skills and innovation capability.
CO4	4, 5, 17	Builds communication, teamwork, and leadership for inclusive, multidisciplinary collaboration and stronger partnerships across diverse teams.
CO5	8, 17	Develops project planning and management skills for successful execution of collaborative professional work.
CO6	4, 8, 16	Cultivates professional ethics, social responsibility, and lifelong learning — reinforcing responsible conduct and good practice in the workplace.

FF No.: 654-A

HS2001: REASONING & APTITUDE DEVELOPMENT- 3

Teaching Scheme	Examination Scheme
Credits : 1	CP: -- Marks
Lectures : 1 Hrs/week	GD/PPT/HA: -- Marks
Practical : -- Hrs/week	MSE (W/O): -- Marks
Tutorial : -- Hrs/week	ESE (W/O): 100 Marks

Prerequisites:	
●	Basic knowledge of English grammar and vocabulary.
●	Fundamental understanding of sentence construction and communication skills.
●	Basic arithmetic and mathematical concepts (Class XII level).
●	Elementary logical and analytical reasoning skills.
●	Familiarity with problem-solving techniques.
●	Basic computer literacy and programming concepts.
●	Exposure to at least one programming language (C/C++/Java/Python) is desirable.
Course Objectives:	
●	Develop proficiency in English grammar, vocabulary, reading comprehension, and effective written communication.
●	Strengthen logical, analytical, critical, and reasoning abilities to solve diverse aptitude-based problems.
●	Apply quantitative and mathematical concepts to solve real-world numerical and analytical problems efficiently.
●	Interpret, analyze, and synthesize information from textual, graphical, tabular, and numerical data for informed decision-making.
●	Develop computational thinking and programming aptitude through logic building, pseudocode, algorithmic problem solving, and debugging techniques.
●	Enhance coding proficiency, optimization skills, and problem-solving abilities to perform effectively in programming assessments, campus recruitment tests, and technical interviews.
Course Outcomes:	
	After completion of the course, student will be able to
1.	**Apply** appropriate English language skills, including grammar, vocabulary, reading comprehension, and effective communication techniques to improve verbal and written proficiency.
2.	**Analyze** logical reasoning problems, critical thinking scenarios, and data interpretation tasks to arrive at accurate and evidence-based conclusions.
3.	**Solve** quantitative aptitude problems using mathematical concepts, algebraic reasoning, probability, geometry, and analytical techniques in real-life contexts.
4.	**Develop** efficient algorithmic solutions by applying computational thinking, programming fundamentals, debugging, optimization techniques, and time complexity analysis to solve coding problems.

Section1:	Topics/Contents

English Language and Communication Skills: English tenses (Present, Past, Future), active and passive voice, sentence correction, parts of speech, syntax, sentence structure, subject–verb agreement, reading comprehension, skimming and scanning, inference, tone, theme, intent identification, contextual vocabulary.

Logical Reasoning and Analytical Thinking: Deductive reasoning (syllogisms, coding–decoding, statement–conclusion, data sufficiency, directional sense, logical sequencing), inductive reasoning (analogies, classification, number series, sequence completion, trend identification), abductive and critical reasoning (assumptions, arguments, cause–effect, logical evaluation, decision making), information gathering, interpretation and synthesis using graphs, charts, tables and multi-source data.

Section2:	Topics/Contents

Quantitative Aptitude and Mathematical Reasoning: Number systems, integers, fractions, decimals, divisibility, HCF and LCM, prime and rational numbers, algebraic expressions, linear equations and inequalities, proportional reasoning, speed–time–distance, profit and loss, percentages, averages, ages, mixtures, permutations and combinations, probability, geometry, spatial reasoning, pattern recognition and quantitative problem solving.

Programming Aptitude and Automata Skills: Logic building, pseudocode, programming fundamentals, basic coding problems, program correctness, execution efficiency, intermediate coding challenges, data structures, algorithms, debugging, optimization techniques, time complexity analysis, industry best coding practices, Automata Pro IT, Automata Pro Max and comprehensive mock assessments.

Text Books:	
1	R. S. Aggarwal, Quantitative Aptitude for Competitive Examinations, Revised and Enlarged Edition. New Delhi, India: S. Chand and Company Ltd., 2017.
2	R. S. Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning. New Delhi, India: S. Chand Publishing.
3	Dr. M. B. Lal and A. Gupta, CSAT Logical Reasoning & Analytical Ability. Agra, India: Upkar Prakashan.
4	G. Laakmann McDowell, Cracking the Coding Interview: 189 Programming Questions and Solutions, 6th ed. Palo Alto, CA, USA: CareerCup, 2015.
Reference Books:	
1	B. H. Dowden, Logical Reasoning. California State University, Sacramento, CA, USA, 2011.
2	501 Challenging Logic and Reasoning Problems, 2nd ed. New York, NY, USA: LearningExpress, LLC, 2005.
3	Logic and Reasoning Problems. LearningExpress Practice Series.
Link: https://drive.google.com/drive/folders/1fXLzuofNhMFq7skonrC0TpCdaCoT7_i0?usp=sharing	

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO		
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	–	–	–	–	–	–	–	–	3	–	2			
CO2	2	2	–	–	–	–	–	–	1	–	2			
CO3	2	2	–	–	–	–	–	–	1	1	2			
CO4	2	2	3	3	3	–	1	–	1	1	3			
Avg	**2**	**2**	**3**	**3**	**3**	**–**	**1**	**–**	**1.5**	**1**	**2.25**			

CO – WK Mapping (Knowledge & Attitude Profiles)

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	–	✓	–	–	–	–	–	✓	✓
CO2	–	✓	–	–	–	–	–	✓	✓
CO3	–	✓	–	–	–	–	–	✓	✓
CO4	–	✓	–	–	–	–	–	✓	✓

CO – WP Mapping (Complex Engineering Problems)

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	–	–	–	–	–	–	–
CO2	✓	–	✓	✓	–	–	–
CO3	✓	–	✓	✓	–	–	–
CO4	✓	–	✓	✓	–	–	✓

CO – EA Mapping (Complex Engineering Activities)

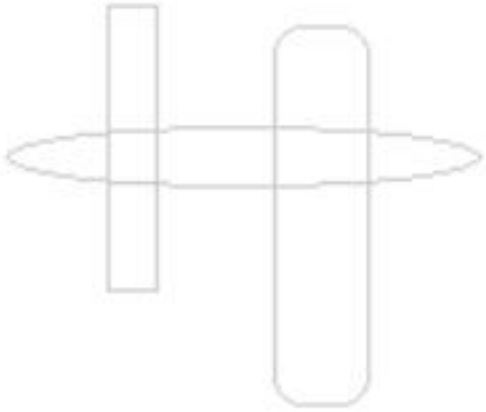
'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	–	–	–	–	–
CO2	–	–	✓	–	✓
CO3	–	–	✓	–	✓
CO4	✓	–	✓	–	✓

CO – SDG Mapping (Sustainable Development Goals)

CO	SDG(s) Addressed	Justification / Linkage
CO1	**SDG 4**	Develops proficiency in English communication, reading comprehension, and language skills, promoting inclusive and quality education and lifelong learning.
CO2	**SDG 4**	Enhances logical reasoning, critical thinking, and analytical decision-making skills, enabling responsible problem-solving and informed decision-making.
CO3	**SDG 4, 8**	Strengthens quantitative aptitude and analytical skills essential for employability, professional competence, and workplace productivity.
CO4	**SDG 4, 8, 9**	Develops computational thinking, programming, and algorithmic problem-solving skills required for innovation, digital transformation, and industry-ready engineering professionals.

- **SDG 4:** Quality Education
- **SDG 8:** Decent Work and Economic Growth
- **SDG 9:** Industry, Innovation and Infrastructure



FF No.: 654-A

DS2007: Design Thinking 1 and 2

Teaching Scheme	Examination Scheme
Credits : 1	CP: Marks
Lectures : - Hrs/week	GD/PPT/HA: Marks
Practical : - Hrs/week	MSE (W/O):- Marks
Tutorial : 1 Hrs/week	ESE (W/O):100 Marks

Prerequisites:	
●	Course Prerequisites: Problem Based Learning, Project Centric Learning Course Relevance:
Course Objectives:	
●	To provide ecosystem for students and faculty for paper publication and patent filing
Course Outcomes:	**[Publication of paper or patent]**
	After completion of the course, student will be able to
1.	Understand the importance of doing Research
2.	Interpret and distinguish different fundamental terms related to Research
3.	Apply the methodology of doing research and mode of its publication
4.	Write a Research Paper based on project work
5.	Understand Intellectual property rights
6.	Use the concepts of Ethics in Research , Understand Entrepreneurship and Business Planning

Section1:	**Topics/Contents**
What is research? Importance of Paper Publication and Patents,Importance of Paper Publication and Patents Structure of Paper ,Journal Publication Publication in conference Literature Review Research Paper Writing. Journal Ratings and Evaluation How to rate a Journal? ,Intellectual property (IP) Research Ethics Entrepreneurship	
Section2:	**Topics/Contents**
Structure of The paper Journal List (Top 50 Journals) Selection of the journal , Use of various online journal selection tools Plagiarism checking ,Improving contents of the paper Patent drafting ,Patent search Filing of patent ,Writing answers to reviewer questions Modification in manuscript ,Checking of publication draft	

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO		
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	1	1	1	1	1							1	2	2
CO2	1	1	1	1	1							1	2	2
CO3	2	2	3	3	2	2	1	2	2	3		1	3	3
CO4	3	3	3	3	3	2	1	2	2	3	1	1	3	3
CO5	1	1	1	1	1							1	2	2
CO6	2	2	2	2	2	2	1	3	2	3		1	3	3
Avg	1.7	1.7	1.8	1.8	1.7	2.0	1.0	2.3	2.0	3.0	1.0	1.0	2.5	2.5

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓		✓					✓
CO2		✓	✓	✓	✓				
CO3		✓	✓	✓	✓	✓			
CO4			✓	✓	✓	✓	✓		
CO5				✓	✓	✓	✓	✓	
CO6	✓			✓		✓		✓	✓

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓	✓	✓		✓		✓
CO2	✓	✓	✓	✓	✓		
CO3	✓	✓	✓	✓	✓	✓	
CO4	✓	✓	✓	✓	✓	✓	✓
CO5	✓	✓	✓	✓	✓	✓	✓
CO6			✓		✓	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	✓			
CO2	✓	✓	✓		
CO3	✓	✓	✓	✓	
CO4		✓	✓	✓	✓
CO5			✓	✓	✓
CO6				✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4, 9	Understanding the importance of research promotes quality education, scientific inquiry, innovation, and technological advancement for solving real-world problems.
CO2	4, 9	Knowledge of research terminology and concepts strengthens analytical thinking, research competency, and innovation capabilities.
CO3	9, 17	Applying research methodology and publishing research fosters innovation, knowledge dissemination, and collaboration among researchers and institutions.
CO4	4, 9, 17	Writing research papers develops scientific communication, promotes lifelong learning, and contributes to global knowledge sharing through collaborative research.
CO5	9,12	Understanding Intellectual Property Rights encourages innovation, protection of inventions, responsible use of knowledge, and sustainable technological development.
CO6	4, 16	Applying research ethics promotes academic integrity, responsible conduct of research, transparency, fairness, and ethical decision-making.
CO7	8, 9	Entrepreneurship and business planning encourage innovation, job creation, sustainable economic growth, and technology-based enterprises.

FF No.: 654-A

DS2008: Engineering Design & Innovation-I

Teaching Scheme	Examination Scheme
Credits : 2	CP: - Marks
Lectures : -- Hrs/week	GD/PPT/HA: - Marks
Practical : 4 - Hrs/week	MSE (R): 30 Marks
Tutorial : -- Hrs/week	ESE (R): 70 Marks

Prerequisites:	
●	Course Prerequisites: Problem Based Learning Course Relevance: Project Centric Learning (PCL) is a powerful tool for students to work in areas of their choice and strengths. Along with course based projects, curriculum can be enriched with semester long Engineering Design and Development courses, in which students can solve socially relevant problems using various technologies from relevant disciplines. The various socially relevant domains can be like Health care, Agriculture, Defense, Education, Smart City, Smart Energy and Swaccha Bharat Abhiyan. To gain the necessary skills to tackle such projects, students can select relevant online courses and acquire skills from numerous sources under guidance of facultyand enrich their knowledge in the project domain, thereby achieving project centric learning. Modern world sustained and advanced through the successful completion of projects. In short, if students are prepared for success in life, we need to prepare them for a project-based world. It is a style of active learning and inquiry-based learning. Project centric learning will also redefine the role of teacher as mentor in the learning process. The PCL model focuses the student on a big openended question, challenge, or problem to research and respond to and/or solve. It brings students not only to know, understand and remember rather it takes them to analyze, design and apply categories of Bloom's Taxonomy.
Course Objectives:	
●	To develop critical thinking and problem solving ability by exploring and proposing solutions to realistic/social problems.
●	To Evaluate alternative approaches, and justify the use of selected tools and methods,
●	To emphasize learning activities those are long-term, inter-disciplinary and student-centric.
●	To engage students in rich and authentic learning experiences.
●	To provide every student the opportunity to get involved either individually or as a group so as to develop team skills and learn professionalism.
●	To develop an ecosystem to promote entrepreneurship and research culture among the students
Course Outcomes:	
	After completion of the course, student will be able to
1.	Identify the real life problem from societal need point of view
2.	Choose and compare alternative approaches to select most feasible one
3.	Analyze and synthesize the identified problem from technological perspective.
4.	Design the reliable and scalable solution to meet challenges.
5.	Evaluate the solution based on the criteria specified
6.	Inculcate long life learning attitude towards the societal problems

Section1:	Topics/Contents

Preamble - The content and process mentioned below is the guideline document for the faculties and students to start with. It is not to limit the flexibility of faculty and students; rather they are free to explore their creativity beyond the guideline mentioned herewith. For all courses of ED, laboratory course contents of "Engineering Design" are designed as a ladder to extend connectivity of software technologies to solve real word problem using interdisciplinary approach. The ladder in the form of gradual steps can be seen as below:

Industry Communication Standards, Single Board Computers and IoT, Computational Biology (Biomedical and Bioinformatics), Robotics and Drone, Industry 4.0 (Artificial Intelligence, Human Computer Interfacing, 5G and IoT, Cloud Computing, Big Data and Cyber Security etc).

Group Structure:

There should be a team/group of 4-5 students. A supervisor/mentor teacher assigned to individual groups. It is useful to group students of different abilities and nationalities together.

Selection of Project/Problem:

● Students must focus on initiating the task/idea. The idea of inception and consideration shall be from following areas as a real world problem: -Health Care, Agriculture, Defense, Education, Smart City, Smart Energy, Swaccha Bharat Abhiyan, Environment, Women Safety.

● This is the sample list to start with. Faculty and students are free to include other ar
eas which meet society's requirements at large.

● The model begins with the identifying of a problem, often growing out of a question or "wondering". This formulated problem then stands as the starting point for learning. Students design and analyze the problem/project within an articulated disciplinary subject frame/domain.

● A problem can be theoretical, practical, social, technical, symbolic, cultural, and/or scientific and grows out of students' wondering within different disciplines and professional environments. A chosen problem has to be exemplary. The problem may involve an interdisciplinary approach in both the analysis and solving phases.

● By exemplarity, a problem needs to refer to a particular practical, scientific, social and/or technical domain. The problem should stand as one specific example or manifestation of more general learning outcomes related to knowledge and/or modes of inquiry.

Teacher's Role in PCL:

● Teacher is not the source of solutions rather he will act as the facilitator and mentor.

● To utilize the principles of problem solving, critical thinking and metacognitive skills of the students.

● To make the group aware about time management.

● Commitment to devote the time to solving student's technical problems and interested in helping students to empower them better.

Student's Role in PCL:

● Students must have ability to initiate the task/idea, they should not be mere imitators.

● They must learn to think.

● Students working in PCL must be responsible for their own learning.

● Students must quickly learn how to manage their own learning, instead of passively receiving instruction.

● Students in PCL are actively constructing their knowledge and understanding of the situation in groups.

● Students in PCL are expected to work in groups.

● They have to develop interpersonal and group process skills, such as effective listening or coping creatively with conflicts.

Developing Inquiry Skills:

- Students in PCL are expected to develop critical thinking abilities by constantly relating: What they read to do? What do they want to do with that information?
- They need to analyze information presented within the context of finding answers.
- Modeling is required so that the students can observe and build a conceptual model of the required processes.
- Use the following mechanism to maintain the track of moving towards the solution.
  - How effective is …….?
  - How strong is the evidence for ……?
  - How clear is ……?
  - What are the justifications for thinking? Why is the method chosen?
  - What is the evidence given to justify the solution?

Literature Survey – To avoid reinvention of wheel:

- It is an integral part of self- directed learning
- Identify the information needed to solve a given problem or issue
- Be able to locate the needed information
- Use the information to solve the given problem effectively.
- Skills required by students in information literacy include:
- · How to prepare for the search? How to carry out research
- Sorting and assessing of information in general

Use of Research Methodology: - investigation, collaboration, comprehension, application, analysis, synthesize and evaluation

Focus on the following skills while working in a team to reach to solution:

- Collaborative learning
- Interpersonal Skills
- Resources Evaluation
- Metacognitive Skills
- Reflection Skills

Section2:	**Topics/Contents**

EDD Sample Case Studies: -

With the adaptation of industry communication standards, Raspberry Pi and Sensors, the following projects can be taken up:

1) Design a deployable product for soil moisture detection
2) Design a deployable product for temperature detection
3) Design a deployable product for pressure detection
4) Design a deployable product smoke detection
5) Design a deployable product for motion detection
6) Design a deployable product for collision detection
7) Design a deployable product for sound detection

...**not limited to.....Faculty and students are free to include other areas which meet society's requirements at large.**

Text Books:

1	A new model of problem based learning. By Terry Barrett. All Ireland Society for higher education (AISHE). ISBN:978-0-9935254-6-9; 2017
2	Problem Based Learning. By Mahnazmoallem, woei hung and Nada Dabbagh, Wiley Publishers. 2019.
3	Stem Project based learning and integrated science, Technology, Engineering and mathematics approach. By Robert Robart Capraro, Mary Margaret Capraro

4	A new model of problem based learning. By Terry Barrett. All Ireland Society for higher education (AISHE). ISBN:978-0-9935254-6-9; 2017
Reference Books:	
1	De Graaff E, Kolmos A., red.: Management of change: Implementation of problembased and project-based learning in engineering. Rotterdam: Sense Publishers. 2007.
2	Project management core textbook, second edition, Indian Edition , by Gopalan.
3	The Art of Agile Development. By James Shore & Shane Warden

List of Practical's:

1.
2.
3.

List of Project areas:

1.
2.
3.

Suggest an assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.
To focus on the higher levels of the Booms Taxonomy analyze, apply, evaluate and create.
MOOCs Links and additional reading material: www.nptelvideos.in https://worldwide.espacenet.com/

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO		
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**	**PSO3**
CO1	2	2	2	2					3		2	3	3	2
CO2	2	2	3	2	2		2		3		2	3	3	2
CO3	2	2	3	2	3		2		3		2	3	3	2
CO4	2	2	3	2	3	3		2	3		2	3	3	2
CO5	2	2	3	2	3	2			3		2	3	3	2
CO6	2	2	3	3	2				3		3	3	3	2
Avg	2.0	2.0	2.83	2.83	2.6	2.5	2.0	2.0	3.0	1.0	2.16	3.0	3.0	2.0

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1		✓	✓	✓	✓			✓	
CO2		✓	✓	✓	✓	✓		✓	
CO3		✓	✓	✓	✓	✓		✓	
CO4		✓	✓	✓	✓	✓	✓	✓	✓
CO5		✓	✓	✓	✓	✓	✓	✓	
CO6		✓	✓	✓	✓	✓		✓	

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓			✓		✓	✓
CO2	✓	✓				✓	
CO3	✓		✓		✓		
CO4	✓				✓	✓	
CO5	✓	✓	✓				
CO6	✓					✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	✓			
CO2	✓	✓	✓	✓	
CO3	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓
CO5	✓	✓	✓	✓	✓
CO6	✓	✓	✓		✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4,9	Identifies real-life societal problem and develops engineering knowledge for innovative solutions
CO2	4,9,12	Evaluates and compares alternative solutions to select sustainable and resource efficient approaches
CO3	4,9,11	Analyze engineering problems and synthesizes technological solutions for societal needs
CO4	4,9,11,13	Designs reliable,scalable and sustainable engineering solutions addressing environmental and societal challenges
CO5	4,9,16	Evaluates engineering solutions based on defined technical, ethical,and quality criteria to ensure reliability and accountability
CO6	4,17	Promotes lifelong learning,professional development,collaboration and commitment to solving societal problems