Savitribai Phule Pune University, Pune

Maharashtra, India

Faculty of Science and Technology

National Education Policy (NEP)-2020 Compliant Curriculum

SE - Second Year Engineering (2024 Pattern) in

Computer Science and Design (CSD)

(With effect from Academic Year 2025-26)

Contents

Preface by Board of Studies	**1**
Program Educational Objectives	**2**
Knowledge and Attitude Profile (WK)	**3**
Program Outcomes	**4**
General Rules	**6**
Curriculum Structure - Semester III	**9**
Curriculum Structure - Semester IV	**10**
Semester III Courses	**11**
Data Structures and Algorithm	**12**
Database Management System	**12**
Discrete Mathematics	**14**
Data Structures and Algorithm Laboratory	**17**
Database Management System Laboratory	**20**
Digital Electronics	**25**
Entrepreneurship Development	**27**
Universal Human Values and Professional Ehtics	**32**
Community Engagement Project	**35**
Semester IV - Courses	**39**
Advance Data Structures	**40**
System Programming and Operating System	**42**
Computer Graphics	**44**
Advance Data Structures Lab	**47**
Computer Graphics Lab	**51**

Computer Organization and Architecture **54**

Object Oriented Programming(Java) **56**

Modern Indian Languages(Marathi) **59**

Engineering Product Design **63**

Environmental Studies **68**

Acknowledgement **68**

Nomenclature

AEC Ability Enhancement Course

AICTE All India Council for Technical Education

CEP Community Engagement Project

EEM Entrepreneurship/Economics/Management Courses

MDM Multidisciplinary Minor

MOOC Massive Open Online Course

NEP National Educational Policy

NPTEL National Programme on Technology Enhanced Learning

OE Open Elective

PCC Program Core Course

PEO Programme Educational Objectives

PSO Program Specific Outcomes

SWAYAM Study Webs of Active-Learning for Young Aspiring Minds

VEC Value Education Course

VEC Value Education Course

VSE Vocational and Skill Enhancement Course

WK Knowledge and Attitude Profile

Preface by Board of Studies

Dear Students and Teachers,

We, the members of Board of Studies Computer Engineering, are very happy to present Second Year Computer Science and Design Engineering syllabus effective from the Academic Year 2025-26. The present curriculum will be implemented for Second Year of Engineering from the academic year 2025-26.Subsequently this will be carried forward for TE and BE in AY 2026-27, 2027-28 respectively. Computer Science and Design Engineering is a dynamic discipline that provides the foundation for the design, development,and application of computer systems and other computing devices. This curriculum is designed to provide students with a comprehensive understanding of the fundamental principles, theories,and practices of computer engineering, while also preparing them for the ever-evolving technological land scape.The revised syllabus falls in line with the objectives of NEP - 2020, Savitribai Phule Pune University, AICTE New Delhi, UGC, and various accreditation agencies by keeping an eye on the technological developments, innovations, and industry requirements. Wherever possible additional resource links of platforms such as NPTEL, Swayam are appropriately provided at the end of each course. Learners are now getting sufficient time for self learning either through online courses or additional projects for enhancing their knowledge and skill sets. This will definitely help learners to facilitate their enhanced learning based on their interest. This curriculum is the result of extensive consultation with academic experts, industry professionals, and alumni to ensure relevance and excellence. It is designed not only to meet the current industry standards but also to prepare students for higher studies and research in the field of computer engineering.We hope that this curriculum will inspire students to become competent professionals, responsible citizens, and contributors to the technological advancement of society.

Dr. Nilesh J. Uke
Chairman, Board of Studies (Computer Engineering), SPPU, Pune

Members of Board of Studies - Computer Engineering	
Dr. Pramod Patil	Dr. Dipti Patil
Dr. Dhananjay Kshirsagar	Dr. Amol Potgantwar
Dr. Sachin Babar	Dr. Balwant Sonkamble
Dr. Suhasini Itkar	Dr. Sachin Sakhare
Dr. Dipak Patil	Dr. Vandana Dhingra
Dr. Deepali Ujalambkar	Dr. Vaishali Vikhe
Dr. Pradip Jawandhiya	Dr. Sandeep Deshmukh

Program Specific Outcomes (PSO)

- **PSO1:** Professional Skills-The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexities.
- **PSO2:** Problem-Solving Skills- The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.
- **PSO3:** Successful Career and Entrepreneurship- The ability to employ modern computer languages, environments and platforms in creating innovative career paths to be an entrepreneur and to have a zest for higher studies.

Programme Educational Objectives (PEO)

Program Educational Objectives (PEOs): Program Educational Objectives are broad statements that describe the career and professional accomplishments that the program is preparing graduates to achieve.

PEO	PEO Focus	PEO Statements
PEO1	Core competence	Attainment of key principles and practices of computation, mathematics and basic principles of engineering to ensure that graduates are able to apply their software development skills in design and implementation of practical systems consisting of software and/or hardware components.
PEO2	Problem solving skills and Ethics	Analyze real-life problems and impart science-based engineering education to develop professional skills that will prepare the students for immediate employment in the industry.
PEO3	Professionalism and Lifelong Learning	Imbibe lifelong learning, professional and ethical attitude for embracing global challenges and make positive impact on environment and society.

Knowledge and Attitude Profile (WK)

A Knowledge and Attitude Profile (KAP), often represented as WK (Knowledge and Attitude Profile) in some contexts, is a framework or assessment tool used to evaluate an individual's knowledge and attitudes related to a specific area, topic, or domain.

WK1	A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.
WK2	Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.
WK3	A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.
WK4	Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.
WK5	Knowledge, including efficient resource use, environmental impacts, whole-life cost, re-use of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.
WK6	Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.
WK7	Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.
WK8	Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.
WK9	Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

Reference: Self-Assessment Report (SAR) Format Undergraduate Engineering Programs Graduate Attributes and Professional Competencies Version 4.0 (GAPC V4.0) - (August 2024) Page 55.

Programme Outcomes (PO)

Program Outcomes are statements that describe what students are expected to know and be able to do upon graduating from the program. These relate to the skills, knowledge, attitude and behaviour that students acquire through the program. On successful completion of B.E. in Artificial Intelligence and Data Science, graduating students/graduates will be able to:

PO1	Engineering knowledge	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	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/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. (WK8)

Reference: Self-Assessment Report (SAR) Format Undergraduate Engineering Programs Graduate Attributes and Professional Competencies Version 4.0 (GAPC V4.0) - (August 2024) Page 56.

General Rules and Guidelines

- Course Outcomes (CO): Course Outcomes are narrower statements that describe what students are expected to know, and are able to do at the end of each course. These relate to the skills, knowledge and behaviour that students acquire in their progress through the course.
- Assessment: Assessment is one or more processes, carried out by the institution, that identify, collect, and prepare data to evaluate the achievement of Program Educational Objectives and Program Outcomes.
- Evaluation: Evaluation is one or more processes, done by the Evaluation Team, for interpreting the data and evidence accumulated through assessment practices. Evaluation determines the extent to which Program Educational Objectives or Program Outcomes are being achieved, and results in decisions and actions to improve the program

Guidelines for Examination Scheme

Theory Examination: The theory examination shall be conducted in two different parts Comprehensive
Continuous Evaluation (CCE) and End-Semester Examination (ESE).
Comprehensive Continuous Evaluation (CCE) :
1. CCE of 30 marks based on all the Units of course syllabus to be scheduled and conducted at institute level.
2. Case studies included under each unit are intended to support applied learning and are part of Comprehensive Continuous Evaluation
3. These case studies will be assessed through internal assessment components such as presentations, assignments, or group discussions. They shall not be included in the End-Semester
Theory Examination.
4. To design a Comprehensive Continuous Evaluation (CCE) scheme for a theory subject of 30 marks with the specified parameters, the allocation of marks and the structure can be detailed as follows:

Sr.	Parameters	Marks	Coverage of Units
1	Unit Test	12 Marks	Units 1 & Unit 2 (6 Marks/Unit)
2	Assignments / Case Study	12 Marks	Units 3 & Unit 4 (6 Marks/Unit)
3	Seminar Presentation / Open Book Test/ Quiz	06 Marks	Unit 5

5. CCE of 15 marks based on all the Units of course syllabus to be scheduled and conducted at institute level. To design a Comprehensive Continuous Evaluation (CCE) scheme for a theory subject of 15 marks with the specified parameters, the allocation of marks and the structure can be detailed as follows:

Sr.	Parameters	Marks	Coverage of Units
1	Unit Test	10 Marks	Units 1 & Unit 2 (5 Marks/Unit)
2	Seminar Presentation / Open Book Test/ Quiz	05 Marks	Units 3 & Unit 4

Format and Implementation of Comprehensive Continuous Evaluation (CCE)

- **Unit Test**
  - **Format :** Questions designed as per Bloom's Taxonomy guidelines to assess various cognitive levels (Remember, Understand, Apply, Analyze, Evaluate, Create).
  - **Implementation**: Schedule the test after completing Units 1 and 2. Ensure the question paper is balanced and covers key concepts and applications.
- **Sample Question Distribution**

- Remembering (2 Marks): Define key terms related to [Topic from Units 1 and 2].
- Understanding (2 Marks): Explain the principle of [Concept] in [Context].
- Applying (2 Marks): Demonstrate how [Concept] can be used in [Scenario].
- Analyzing (3 Marks): Compare & contrast [Two related concepts] from Units 1 and 2.
- Evaluating (3 Marks): Evaluate the effectiveness of [Theory/Model] in [Situation].

- **Assignments / Case Study** : Students should submit one assignment or one Case Study Report based on Unit 3 and one assignment or one Case Study Report based on Unit 4.
  - **Format:** Problem-solving tasks, theoretical questions, practical exercises, or case studies that require in-depth analysis and application of concepts.
  - **Implementation:** Distribute the assignments or case study after covering Units 3 and 4. Provide clear guidelines and a rubric for evaluation.
- **Seminar Presentation:**
  - **Format:** Oral presentation on a topic from Unit 5, followed by a Q&A session.
  - **Deliverables:** Presentation slides, a summary report in 2 to 3 pages, and performance during the presentation.
  - **Implementation:** Schedule the seminar presentations towards the end of the course. Provide students with ample time to prepare and offer guidance on presentation skills.
- **Open Book Test:**
  - **Format:** Analytical and application-based questions to assess depth of understanding.
  - **Implementation:** Schedule the open book test towards the end of the course, ensuring it covers critical aspects of Unit 5.
- **Quiz :**
  - **Format:** Quizzes can help your students practice existing knowledge while stimulating interest in learning about new topic in that course. You can set your quizzes to be completed individually or in small groups.
  - **Implementation:** Online tools and software can be used create quiz. Each quiz is made up of a variety of question types including multiple choice, missing words, true or false etc
- **Example Timeline for conducting CCE:**
  - Weeks 1-4 : Cover Units 1 and 2
  - Week 5 : Conduct Unit Test (12 marks)
  - Weeks 6-8 : Cover Units 3 and 4
  - Week 9 : Distribute and collect Assignments / Case Study (12 marks)
  - Weeks 10-12 : Cover Unit 5
  - Week 13 : Conduct Seminar Presentations or Open Book Test or Quiz (6 marks)
- **Evaluation and Feedback:**
  - **Unit Test:** Evaluate promptly and provide constructive feedback on strengths and areas for improvement.
  - **Assignments / Case Study:** Assess the quality of submissions based on the provided rubric. Offer feedback to help students understand their performance.

- **Seminar Presentation:** Evaluate based on content, delivery, and engagement during the Q&A session. Provide feedback on presentation skills and comprehension of the topic.
- **Open Book Test**: Evaluate based on the depth of analysis and application of concepts. Provide feedback on critical thinking and problem-solving skills.

End-Semester Examination (ESE)

End-Semester Examination (ESE) of 70 marks written theory examination based on all the unit of course syllabus scheduled by university. Question papers will be sent by the University through QPD (Question Paper Delivery). University will schedule and conduct ESE at the end of the semester.

- **Format and Implementation :**
  - **Question Paper Design** : Below structure is to be followed to design an End-Semester Examination (ESE) for a theory subject of 70 marks on all 5 units of the syllabus with questions set as per Bloom's Taxonomy guidelines and 14 marks allocated per unit.
  - **Balanced Coverage**: Ensure balanced coverage of all units with questions that assess different cognitive levels of Bloom's Taxonomy: Remember, Understand, Apply, Analyze, Evaluate, and Create. The questions should be structured to cover:
    - Remembering: Basic recall of facts and concepts.
    - Understanding: Explanation of ideas or concepts.
    - Applying: Use of information in new situations.
    - Analyzing: Drawing connections among ideas.
    - Evaluating: Justifying a decision or course of action.
    - Creating: Producing new or original work (if applicable).
  - **Detailed Scheme for 70 Marks :** Unit-Wise Allocation (14 Marks per Unit): Each unit will have a combination of questions designed to assess different cognitive levels. By following this scheme, you can ensure a comprehensive and fair assessment of students' understanding and application of the course material, adhering to Bloom's Taxonomy guidelines for cognitive skills evaluation
  - **Detailed Scheme for 35 Marks :** Unit-Wise Allocation (08 Marks for Unit 1 , 09 Marks for Unit 2, Unit 3 and Unit 4) : Each unit will have a combination of questions designed to assess different cognitive levels. By following this scheme, you can ensure a comprehensive and fair assessment of students' understanding and application of the course material, adhering to Bloom's Taxonomy guidelines for cognitive skills evaluation.

Second Year Engineering (2024 Pattern) – Computer Science and Design

Course Code	Course Type	Course Name	Teaching Scheme			Examination Scheme						Credits			
			Theory	Tutorial	Practical	CCE	EndSem	Term Work	Practical	Oral	Total	Theory	Tutorial	Practical	Total
PCC-201-CSD	Program Core Course	Data Structures and Algorithms	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC-202-CSD	Program Core Course	Database Management System	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC-203-CSD	Program Core Course	Discrete Mathematics	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC-204-CSD	Program Core Course	Data Structures and Algorithms Laboratory	-	-	4	-	-	25	50	-	75	-	-	2	2
PCC-205-CSD	Program Core Courses	Database Management System Laboratory	-	-	2	-	-	25	25	-	50	-	-	1	1
	Open Elective	*Open Elective-I	2	-	-	15	35	-	-	-	50	2	-	-	2
MDM-221-CSD	Multi Disciplinary	Digital Electronics	2	-	-	30	70	-	-	-	100	2	-	-	2
EEM-231-CSD	Entrepreneurship/ Management	Entrepreneurship Development	-	1	2	-	-	25	-	-	25	-	1	1	2
VEC-232-CSD	Value Education	Universal Human Values and Professional Ethics	2	-	-	15	35	-	-	-	50	2	-	-	2
CEP-241-CSD	Community Engagement Project	Community Engagement Project	-	-	4	-	-	25	-	25	50	-	-	2	2
Total			**15**	**01**	**12**	**150**	**350**	**100**	**75**	**25**	**700**	**15**	**01**	**06**	**22**

*__Note:__ Students can opt for Open Electives offered by different faculty like Arts, Science, Commerce ,Management, Humanities or Inter-Disciplinary studies.

• Example – Open Elective I - Financial Accounting, Digital Finance, Digital Marketing can be opted from Commerce and Management faculty.

• Elective II - Project Management, Business Analytics, Financial Management can be opted from InterDisciplinary studies,Commerce and Management faculty respectively.

Second Year Engineering (2024 Pattern) – Computer Science and Design

Course Code	Course Type	Course Name	Teaching Scheme			Examination Scheme						Credits			
			Theory	Tutorial	Practical	CCE	EndSem	Term Work	Practical	Oral	Total	Theory	Tutorial	Practical	Total
PCC-251-CSD	Program Core Course	Advance Data Structures	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC-252-CSD	Program Core Course	System Programming and Operating System	2	-	-	30	70	-	-	-	100	2	-	-	2
PCC-253-CSD	Program Core Course	Computer Graphics	3	-	-	30	70	-	-	-	100	3	-	-	3
PCC-254-CSD	Program Core Course	Advance Data Structures Laboratory	-	-	2	-	-	25	25	-	50	-	-	1	1
PCC-255-CSD	Program Core Course	Computer Graphics Laboratory	-	-	2	-	-	25	-	25	50	-	-	1	1
	Open Elective	*Open Elective-II	2	-	-	15	35	-	-	-	50	2	-	-	2
MDM-271-CSD	Multi Disciplinary Minor	Computer Organization and Architecture	2	-	-	30	70	-	-	-	100	2	-	-	2
VSE-281-CSD	Vocational and Skill Enhancement Course	Object Oriented Programming (JAVA)	-	-	4	-	-	25	25	-	50	-	-	2	2
AEC-282-CSD	Ability Enhancement Course	Modern Indian Language (Marathi)	-	1	2	-	-	25			25		1	1	2
EEM-283 CSD	Entrepreneurship/ Management	Engineering Product Design	-	1	2	-	-	25	-		25	-	1	1	2
VEC-284 CSD	Value Education Course	Environmental Studies	2	-	-	15	35	-	-	-	50	2	-	-	2
Total			**14**	**02**	**12**	**150**	**350**	**125**	**50**	**25**	**700**	**14**	**02**	**06**	**22**

***Note:** Students can opt for Open Electives offered by different faculty like Arts, Science, Commerce ,Management, Humanities or Inter-Disciplinary studies.

• Example – Open Elective I - Financial Accounting, Digital Finance, Digital Marketing can be opted from Commerce and Management faculty.

• Elective II - Project Management, Business Analytics, Financial Management can be opted from InterDisciplinary studies, Commerce and Management faculty respectively.

Savitribai Phule Pune University, Pune

Maharashtra, India

SE - Computer Science and Design

2024 Pattern

Semester III

With effect from Academic Year 2025-26

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-201-CSD :Data Structures and Algorithm		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Programming and Problem Solving
Companion Courses: Data Structures and Algorithm Laboratory
Course Objectives: The course aims to:

1.To understand the standard and abstract data representation methods.
2.To acquaint with the structural constraints and advantages in usage of the data.
3.To understand various data structures, operations on it and the memory requirements
4.To understand various data searching and sorting methods.
5.To understand various algorithmic strategies to approach the problem solution

.**Course Outcomes:** Upon successful completion of this course, students will be able to:

CO1: Design the algorithms to solve the programming problems, **identify** appropriate algorithmicstrategy for specific application, and **analyze** the time and space complexity.
CO2: Discriminate the usage of various structures, **Design/Program/Implement** the appropriate data structures; use them in implementations of abstract data types and Identity the appropriate data structure in approaching the problem solution.
CO3: Demonstrate use of sequential data structures- Array and Linked lists to store and process data.
CO4: Understand the computational efficiency of the principal algorithms for searching and sorting andchoose the most efficient one for the application.
CO5: Apply principles of data structures-stack and queue to solvecomputational problems.

Unit I - Introduction to Algorithm and Data Structures (09 Hours)

Basics Concepts-Data, Data objects, Data types, Data structure, Abstract Data Types (ADT), Primitive and non-primitive, linear and nonlinear, static and dynamic, persistent and ephemeral data structures. **Algorithms-**Introduction, Characteristics, Analysis of algorithms Complexity of algorithms- Space complexity, Time complexity, Asymptotic notation.

Algorithmic Strategies- Introduction to Algorithm design strategies- Divide and Conquer, and Greedy strategy.

UnitII- Sequential Organization and Searching(09 Hours)

Sequential Organization- Concept, Array as an abstract data type, Memory representation and address calculation, Inserting and deleting an element, Concept of Ordered List.

Searching: Search Techniques-Sequential Search/Linear Search, Variant of Sequential Search- SentinelSearch, Binary Search, Fibonacci Search, and Indexed Sequential Search.

Unit III - Sorting (09Hours)

Sorting: Types of Sorting -Internal and External Sorting, General Sort Concepts-Sort Order, Stability, Efficiency, and Number of Passes, Comparison Based Sorting Methods-Bubble Sort, Insertion Sort, Selection Sort, Quick Sort, Shell Sort, Non-comparison Based Sorting Methods-Radix Sort, Counting Sort, and Bucket Sort, Comparison of All Sorting Methods and their complexities.

Unit IV - Linked List (09 Hours)

Linked lists-Concept, Linked list as an Abstract data type, Comparison of sequential and linked organizations Realization of Linked list- using arrays, using dynamic memory management, header node, advantages and disadvantages of linked list Linked list operations-Insert a node, delete a node, traverse, copy, reverse, concatenate, delete list Types of linked list-Linear, circular , Doubly linked list and operations, introduction to GLL

Unit V - Stack & Queue (09 Hours)

Stacks-Concept, Stack as an ADT, Representation of stacks using array and linked list, stack operations, Multi-stacks Applications of Stack- Polish notation, expression conversion and evaluation, Processing of function calls and Returns Recursion- Concept, Types of Recursion-Direct recursion. Queues- Concept, Queue as ADT, Representation of queues using arrays and linked list, Circular queue, Dequeue, Multi-queues, Linked queue and operations. Applications of Queue: Scheduling, Josephus problem

Learning Resources

Text Books:

1. 1. Bradley N. Miller, David L. Ranum., "Problem Solving with Algorithms and Data Structures using Python", Franklin Beedle Publishers
2. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, "Data Structures and algorithm in Python", Wiley Publication, ISBN: 978-1-118-29027-9
3. Horowitz, Sahani and Mehta , "Fundamentals of Data Structures in C++", University Press, ISBN 10: 0716782928 ISBN 13: 9780716782926

Reference Books

1. Steven S S. Skiena, "The Algorithm Design Manual", Springer, 2nd ed. 2008 Edition, ISBN-13:978-1849967204.
2. Allen Downey, Jeffery Elkner, Chris Meyers, "How to think like a Computer Scientist: Learningwith Python", Dreamtech Press, ISBN: 9789351198147.
3. Brassard and Bratley, "Fundamentals of Algorithmic", Prentice Hall India/Pearson Education, ISBN 13-9788120311312.

MOOC / NPTEL/YouTube Links:

1.https://nptel.ac.in/courses/106/102/106102064/

2.https://nptel.ac.in/courses/106/105/106105085

3.https://nptel.ac.in/courses/106/106/106106127

e-Books:

1.https://www.ebooks.com/en-us/book/95777110/Python-data-structures-and-algorithms/benjamin-baka/

2.https://www.ebookphp.com/advanced-data-structures-epub-pdf/

3.https://www.ebookphp.com/data-structures-and-algorithms-professional-edition-beginners-guide-epub-pdf//

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-202-CSD: Database Management System		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Data Structures
Companion Courses: DBMS Laboratory
Course Objectives: The course aims to:

1. To Understand the fundamental concepts of Database Management Systems
2. To Apply SQL and PL/SQL for database creation, manipulation, querying, and manage advanced database programming features
3. To Analyze database schemas for efficient design and apply normalization techniques up to BCNF,
4. To Comprehend transaction management concepts including ACID properties, concurrency control techniques, and recovery methods
5. To Explore the concepts and types of NoSQL databases, understand the CAP theorem, and compare NoSQL with RDBMS.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Explain Fundamentals of databases systems, Data Models and ER model
- CO2: Implement database queries using SQL and PL/SQL.
- CO3: Apply normalization techniques to optimize database design.
- CO4: Explain transaction management and recovery mechanisms in databases.
- CO5: Apply advanced database concepts.

Course Contents

Unit I - Introduction to DBMS & ER Model (09 Hours)

Fundamentals of DBMS: Purpose of Database Systems, Database-System Applications, View of Data (Physical, Logical, Conceptual), Database Languages (DDL, DML, DCL, TCL), Database System Structure. Data Models & ER Model: Types of Data Models: Hierarchical, Network, Relational, Entity-Relationship (ER) Model: Entities, Attributes, Relationships, Constraints ER Diagram & Design Process Extended ER Features (Specialization, Generalization, Aggregation), Converting ER and EER diagrams into tables. Relational Model & Integrity Constraints: Basic Concepts: Attributes, Domains, Tuples, CODD's Rules for Relational Databases, integrity Constraints: Domain Integrity, Referential Integrity, Enterprise Constraints

Unit II - SQL and PL/SQL (09 Hours)

Introduction to SQL: Characteristics and advantages of SQL Data Types and Literals SQL Commands & Operators: Data Definition Language (DDL): CREATE, ALTER, DROP, TRUNCA, TE, RENAME, Data Manipulation Language (DML): INSERT, UPDATE, DELETE, Data Control Language (DCL): GRANT, REVOKE, Transaction Control Language (TCL): COMMIT, ROLLBACK, SAVEPOINT, SQL Operators: Arithmetic, Comparison, Logical, Set Operators, Tables & Constraints: Creating, Modifying, Deleting Tables, Integrity Constraints: PRIMARY KEY, FOREIGN KEY, UNIQUE, CHECK, NOT NULL Views

& Indexes: Creating, Dropping, Updating Views, Indexes: Types and Usage SQL Queries: SELECT Query & Clauses: , Tuple Variables, Aggregate Functions. Advanced SQL Features: Nested Queries & Subqueries, Database Modification: INSERT,UPDATE, DELETE Queries, Stored Procedures & Functions, Triggers: Row-level & Statement-level, Cursors: Implicit & Explicit, Embedded SQL: Using SQL within programming languages (Java, Python, C) PL/SQL & Advanced SQL Features: Views: Creating, Dropping, Updating using Indexes, Stored Procedures and Functions, Cursors & Triggers, Assertions, Roles, and Privileges

Unit III - Relational Database Design & Normalization (09Hours)

Features of Good Database Design: Atomic Domains & First Normal Form (1NF), Functional Dependencies. Normalization & Decomposition: 2NF, 3NF, BCNF, Decomposition using Functional Dependencies, Algorithms for Decomposition. Introduction to Transactions & ACID Properties: Database Transactions, Transaction States & ACID Properties

Unit IV - Transaction Management & Recovery (09 Hours)

Schedules and Serializability: Concept of Schedules & Serial Schedules, Serializability: Conflict & View Serializability, Cascaded Aborts, Recoverable & Non-Recoverable Schedules. Concurrency Control Mechanisms: Lock-Based Proto cols: Timestamp-Based Protocols, Deadlock Handling. Database Recovery Methods Shadow Paging ,Log-Based Recovery, Checkpoints, Deferred & Immediate Database Modifications

Unit V - NoSQL Databases & Advanced Database Technology (09 Hours)

NoSQL Databases & CAP Theorem: Introduction to Distributed Databases, Advantages & Disadvantages, CAP Theorem ,Types of Data: Structured, Unstructured, Semi-Structured, NoSQL Concepts & Types: Need for NoSQL, Features of NoSQL Types of NoSQL Databases: Key-Value Store, Document Store, Graph Database, Wide Column Stores, BASE vs. ACID Properties, Comparative Study: RDBMS vs. NoSQL, Advanced Database Technologies: Semi-Structured Data: JSON & XML, Object-Relational Database System (ORDBMS),Spatial Databases: Geographic & Geometric Data

Learning Resources

Text Books:

1. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 6th edition
2. Connally T, Begg C., "Database Systems", Pearson Education, ISBN 81-7808-861-4
3. Pramod J. Sadalage and Martin Fowler, “NoSQL Distilled”, Addison Wesley, ISBN- 10: 0321826620, ISBN-13: 978-0321826626

Reference Books:

1. C J Date, “An Introduction to Database Systems”, Addison-Wesley, ISBN: 0201144719
2. S.K.Singh, “Database Systems: Concepts, Design and Application”, Pearson Education, ISBN 978-81-317-6092-5
3. Kristina Chodorow, Michael Dierolf, “MongoDB: The Definitive Guide”, O‘Reilly Publications, ISBN: 978-1-449-34468-9
4. Adam Fowler, “NoSQL For Dummies”, John Wiley & Sons, ISBN-1118905628
5. Kevin Roebuck, “Storing and Managing Big Data - NoSQL, HADOOP and More”, Emereopty Limited, ISBN: 1743045743, 9781743045749

6. Joy A. Kreibich, "Using SQLite", O'REILLY, ISBN: 13:978-93-5110-934-1

7. Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publications ISBN: 9788176569644, 9788176569644

MOOC / NPTEL/YouTube Links: -

1. http://www.nptelvideos.com/lecture.php?id=6518

2. https://www.youtube.com/playlist?list=PLUuRBdEckpS9R5fYvxD6aL2dLk3_n2sPb

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-203-CSD: Discrete Mathematics		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester :** 70 Marks

Prerequisite Courses, if any : Students should have prior knowledge of

1. Basic Mathematics

Course Objectives: The course aims to introduce several Discrete Mathematical Structures found to be serving as tools even today in the development of theoretical computer science.

1. To introduce students to understand, explain, and apply the foundational mathematical concepts at the core of computer science.
2. To understand use of set, function and relation models to understand practical examples, and interpret the associated operations and terminologies in context.
3. To acquire knowledge of logic and proof techniques to expand mathematical maturity.
4. To learn the fundamental counting principle, permutations, and combinations.
5. To study how to model problems using graphs and trees.
6. To learn algebraic structures

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: **Apply** and **Analyze** Set Theory and Propositional Logic
- CO2: **Evaluate** and **Construct** Models using Relations and Functions
- CO3: **Design** and **Implement** Tree Structures and Network Flow Algorithms
- CO4: **Analyze** and **Develop** Solutions using Graph Theory
- CO5: **Apply** and **Solve** Problems using Counting Principles, **Understand** Algebraic

Course Contents

Unit I - Set and Propositions (09 Hours)

Introduction and significance of Discrete Mathematics, Propositional Logic- logic, Propositional Equivalences, Application of Propositional Logic- Translating English Sentences, Proof by Mathematical Induction and Strong Mathematical Induction. Sets– Naïve Set Theory (Cantorian Set Theory), Axiomatic Set Theory, Set Operations, Cardinality of set, Principle of inclusion and exclusion. Types of Sets – Bounded and Unbounded Sets, Diagonalization Argument, Countable and Uncountable Sets, Finite and Infinite Sets, Countably Infinite and Uncountably Infinite Sets, Power set.

Case study: Know about the great philosophers- Georg Cantor, Richard Dedekind and Aristotle. Design a recommendation system using logical propositions and predicates to filter movies based on user preferences.

Unit II - Relations and Functions (09 Hours)

Introduction to Relations and their Properties Representation of Relations using Matrices and Digraphs Equivalence relations, Partial orderings, Partitions, Hasse diagram, Lattices, Chains and Anti-Chains, Transitive closure and Warshall's algorithm.

Functions: Types of Functions (Injective, Surjective, Bijective) , Composition and Inverse of Functions , Recursive Functions and Applications in Algorithms, Counting Functions and Growth of Functions

Cast Study - Know about the great philosophers-Dirichlet

Unit III - Introduction to Trees (09 Hours)

Introduction to Trees and Properties, Binary Trees and Binary Search Trees (BST), Tree Traversal Techniques: Preorder, Inorder, Postorder , Huffman Trees and Data Compression Algorithms, Applications of Trees in File Systems, The Max flow- Min Cut Theorem in Transport network.

Case Studies - Algebraic Expression Tree, Tic-Tac-Toe Game Tree, implement a file directory system using a tree structure, allowing hierarchical organization of files and folders

Unit IV - Introduction to Graph Theory (09 Hours)

Introduction to Graphs: Types and Representation, Graph Traversals: BFS and DFS ,Connected Components and Path finding Algorithms, Eulerian and Hamiltonian Paths and Circuits , Planar Graphs and Graph Coloring, Dijkstra's Algorithm for Shortest Paths , Spanning Trees and Minimum Spanning Tree Algorithms (Prim's and Kruskal's)

Case study : Model a social media platform using directed graphs to represent relationships such as "follower" or "friend." Three utility problem, Web Graph, Google map

Unit V - Counting Principles and Algebraic Structures - (09 Hours)

Basic Counting Techniques: Addition and Multiplication Principles, Permutations and Combinations, Binomial Coefficients and Pascal's Triangle, Pigeonhole Principle and its Applications, Inclusion-Exclusion Principle, Generating Functions for Counting Problems.

The structure of algebra - Algebraic Systems, Semi Groups, Monoids, Groups, Homomorphism and Normal Subgroups and Congruence relations, Rings, Integral Domains and Fields.

Case Studies - Study Sudoku solving algorithms and algorithm for generation of new SUDOKU. Study Hank-shake Puzzle and algorithm to solve it Calculate the number of possible password combinations given specific constraints on length, character types, and repetition

Learning Resources

Text Books:

1. Kenneth H. Rosen, "Discrete Mathematics and its Applications", Tata McGraw-Hill, ISBN 978-0-07-288008-3
2. Bernard Kolman, Robert C. Busby and Sharon Ross, "Discrete Mathematical Structures", Prentice-Hall of India /Pearson, ISBN: 0132078457, 9780132078450.
3. Narsingh Deo, "Graph with application to Engineering and Computer Science", Prentice Hall of India, 1990, 0 – 87692 – 145 – 4.
4. Eric Gossett, "Discrete Mathematical Structures with Proofs", Wiley India Ltd, ISBN:978-81-265-2758-8.
5. Sriram P.and Steven S., "Computational Discrete Mathematics", Cambridge University Press, ISBN 13: 978-0-521-73311-3.
6. Herstein, I. N. Topics in Algebra. 2nd ed., Indian Adaptation, Wiley India Pvt. Ltd., 2006. ISBN: 9788126510184.

Reference Books:

1. https://www.ebookphp.com/discrete-mathematical-structures-6th-edition-epub-pdf/

2. http://discrete.openmathbooks.org/pdfs/dmoi-tablet.pdf
3. http://home.iitk.ac.in/∼arlal/book/mth202.pdf
4. https://web.stanford.edu/class/cs103x/cs103x-notes.pdf
5. http://home.iitk.ac.in/∼arlal/book/mth202.pdf

MOOC / NPTEL/YouTube Links: -

1. https://www.nptel.ac.in/courses/106/106/106106094/
2. https://nptel.ac.in/courses/106/106/106106183/
3. https://nptel.ac.in/courses/106/103/106103205/
4. https://nptel.ac.in/courses/106/105/106105192/
5. https://nptel.ac.in/courses/111/106/111106050/

E-Books :

1. https://www.ebookphp.com/discrete-mathematical-structures-6th-edition-epub-pdf/
2. http://discrete.openmathbooks.org/pdfs/dmoi-tablet.pdf
3. http://home.iitk.ac.in/∼arlal/book/mth202.pdf
4. https://web.stanford.edu/class/cs103x/cs103x-notes.pdf
5. http://home.iitk.ac.in/∼arlal/book/mth202.pdf

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-204-CSD: Data Structures and Algorithms Laboratory		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical : 04 Hours/Week	02	**Term Work** : 25 Marks **Practical :** 50 Marks

Prerequisite Courses : Basics of python programming and Principles of Problem Solving
Companion Course : Data Structures and Algorithm
Course Objectives:
To understand basic techniques and strategies of algorithm analysis, the memory requirement for various data structures like array, linked list, stack, Queue etc. using concepts of python programming language.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Demonstrate algorithms on various linear data structure using sequential organization to solve real life problems.
- CO2: Apply suitable searching and sorting algorithm to various applications.
- CO3: Analyze problems to use variants of linked list and solve various real life problems.

Course Contents

Guidelines for Instructor's Manual

The instructor's manual/Lab Manual is to be developed as a hands-on resource and reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of course, conduction and Assessment guidelines, topics under consideration-concept, objectives, outcomes, set of typical applications/assignments/guidelines, references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software and Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis. Program codes with sample output of all performed assignments are to be submitted as softcopy.

As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal may be avoided. Students programs maintained on cloud or college server by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory for accreditation purpose.

Guidelines for Laboratory/Term Work Assessment

Continuous assessment of laboratory work should be done based on overall performance and Laboratory assignments performance of student. Each Laboratory assignment assessment should be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include timely completion performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students.

The instructor may set multiple sets of assignments and distribute among batches of students. It is appreciated if the assignments are

Based on real world problems/applications. Encourage students for appropriate use of Hungarian notation,

Proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned. Instructor may also set one assignment or mini-project that is suitable to respective branch beyond the scope of syllabus.

Assignments should be implemented in Python without using built-in methods for major functionality of assignment. Use List data structure of Python as array.

Each student must perform at least 10 assignments)

Operating System Recommended: - 64-bit Open source Linux or its derivative Programming tools recommended: - Open Source Python, Programming tool like Jupiter Notebook, Pycharm, Spyder

Guidelines for Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student's for advanced learning, understanding of the fundamentals, effective and efficient implementation. So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Suggested List of Laboratory Experiments/Assignments

1	Write a C program to store marks scored in subject "Data Structures and Algorithms" by N students in the class. Write functions to compute following: a.The Avegare score of class b.Highest score & lowest score of class c.Count of students who were absent for test d.Display marks with highest frequency
2	Write a C program for department library which has N books, write functions for following: a.Delete the duplicate entries b.Display books in ascending order based on cost of books c.Count number of books with cost more than 500. d.Copy books in a new list which has cost less than 500.
3	Write a C program to compute following operations on String: a.To display word with the longest length b.To determines the frequency of occurrence of particular character in the string c.To check whether given string is palindrome or not d.To display index of first appearance of the substring e.To count the occurrences of each word in a given string
4	Write a C program to maintain club members, sort on roll numbers in ascending order. Write function "Ternary Search" to search whether particular student is member of club or not. Ternary search is modified binary search that divides array into 3 halves instead of two.
5	Write a Python program to store first year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using a.Selection Sort b.Bubble sort and display top five scores.

6	Write a Python program to store second year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using a.Insertion sort b.Shell Sort and display top five scores
7	Write a Python program to store first year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using quick sort and display top five Scores.
8	Write a Python program to store 12th class percentage of students in array. Write function for sorting array of floating point numbers in ascending order using bucket sort and Display top five scores.
9	Write a Python program to implement Stacks to checking for balanced parentheses in an expression
10	Write a Python program for Queue Implementation with Application: Printer Job Queue.
11	Write a Python program that implements a Singly Linked List, using the Music Playlist Management as the real-world application. It will allow us to add, remove, and display songs in a playlist.
12	Write a Python program that implements an Inventory Management System using a Doubly Linked List. This system allows us to add, remove, and display products in an inventory, with the ability to manage products from both the front and end of the list.

Learning Resources

Text Books

1. Data structures and algorithms in python by Michael T. Goodrich, ISBN-13: 978- 1118290279, ISBN-10: 1118290275, Publisher: Wiley; 1st edition (March 18, 2013).
2. Problem Solving with Algorithms and Data Structures Using Python by Bradley N Miller and David L. Ranum. ISBN-13: 978-1590282571, ISBN-10: 1590282574, Publisher: Franklin, Beedle & Associates; 2nd edition (August 22, 2011).

Reference Books

1. Hands-On Data Structures and Algorithms with Python: Write complex and powerful code using the latest features of Python 3.7, 2nd Edition by Dr. Basant Agarwal, Benjamin Baka. ISBN: 9781788991933, 2018.
2. Core Python Programming -R. Nageswara Rao, ISBN-10: 9789351199427, ISBN-13: 978-9351199427, Willy; 1st edition (January 1, 2016).

MOOC/NPTEL/SWAYAM Course Links:

1. NPTEL :- Programming, Data Structures and Algorithms using Python By Prof. Madhavan Mukund, Chennai Mathematical Institute, https://archive.nptel.ac.in/courses/106/106/106106145/

YouTube/Video Links:

1. https://www.youtube.com/playlist?list=PLeo1K3hjS3uu_n_a__MI_KktGTLYopZ12

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-205-CSD:Database Management System Laboratory		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	**Term Work** : 25 Marks **Practical :** 25 Marks

Prerequisite Courses, if any :

1. Data Structures Lab

Companion Course: Database Management System

Course Objectives: The course aims to:

- To understand the fundamentals of database management System and database query languages
- To know the principles of database design and transaction management
- To study database system architecture and NOSQL databases

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Make use of normalized relational database schemas to represent real- world scenarios
- CO2: Build simple and complex SQL queries and PL/ SQL code to retrieve, manipulate relational database
- CO3: Construct ER diagram to represent logical design of a database
- CO4: Build database queries using MongoDB to retrieve, manipulate NoSQL databases
- CO5: Develop database-driven applications using programming languages and frameworks that interact with relational database systems or NoSQL databases

Course Contents

Guidelines for Instructor's Manual

The instructor's manual/Lab Manual is to be developed as a hands-on resource and reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of course, conduction and Assessment guidelines, topics under consideration-concept, objectives, outcomes, set of typical applications/assignments/guidelines, references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software and Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis. Program codes with sample output of all performed assignments are to be submitted as softcopy.

As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal may be avoided. Students programs maintained on cloud or college server by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory for accreditation purpose.

Guidelines for Laboratory/Term Work Assessment

Continuous assessment of laboratory work should be done based on overall performance and Laboratory assignments performance of student. Each Laboratory assignment assessment should be assigned

grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include timely completion performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy needs to address the average students and inclusive of an element to attract and promote the intelligent students. The instructor may set multiple sets of assignments and distribute them among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructors may assign one real life application in the form of a mini-project based on the concepts learned. Instructors may also set one assignment or mini-project that is suitable to respective branch beyond the scope of the syllabus.

Operating System recommended :- 64-bit Open source Linux or its derivative Programming tools recommended: - MYSQL/Oracle, MongoDB, ERD plus, ER Win

Each student must perform at least 08 assignments)

Guidelines for Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student's for advanced learning, understanding of the fundamentals, effective and efficient implementation. So encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. So adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Suggested List of Laboratory Experiments/Assignments

Sr.	Name of Assignment

1	Consider the given Database Schema: student (student-name, street, city) enrolls(student-name,college-name,scholarship) college (college-name, city) mentors (student-name, mentor-name) Write SQL queries for the following 1.Find the names of all students who are enrolled in First National College. 2.Find the names and cities of residence of all students who are enrolled in First National College. 3.Find the names, street addresses, and cities of residence of all students who are enrolled in First National College and receive a scholarship of more than Rs. 10,000. 4.Find all students in the database who live in the same cities as the colleges in which they are enrolled. 5.Find all students in the database who live in the same cities and on the same streets as do their mentors. 6.Find all students in the database who are not enrolled in First National College. 7.Find all students in the database who receive a scholarship greater than each student of Small Town College. 8.Assume that colleges may be located in several cities. Find all colleges located in every city in which Small Town College is located. 9.Find all students who receive a scholarship greater than the average scholarship of all students at their college. 10.Find the college that has the most students enrolled. 11.Find the college that has the smallest total scholarship payout. 12.Find those colleges whose students receive a higher average scholarship than the average scholarship at First National College
2	Consider the given relational table: Student (stdno, stdname, course, city, scholarship, zipcode, district) Write SQL queries for the following: 1.Create a sequence used to generate student numbers for the stdno column of the student table. 2.Create an Index on the district. 3.Find the district whose zip code = 071 and check whether the query uses the Index and write your observation. 4.Create a view for students having scholarship < 50000 and residing in 'Mumbai'. 5.Display a count of students who reside in 'Mumbai'. 6.Find the average scholarship of students from the created view. 7.Display student names who reside on the same street as the students in the view.
3	**SQL Joins** Consider the given database schema: -Patient (patientid, patientname, doctorid, patientcity) -Doctor (doctorid, doctorname, doctorcity, specialization) Use all types of Joins and write SQL queries for the following: 1.Find the doctor of each patient. 2.Find the patients who are not assigned to any doctor. 3.Find the patients who are not assigned to any doctor as well as doctors who do not have any patients. 4.Find the patients whose doctor's specialization is Cardiology. 5.Create a view containing the total number of patients whose doctor belongs to Mumbai.

4	**ER Modelling and Normalization:** Design using ER features using tools like ERD plus, ER Win etc. (Identifying entities, relationships between entities, attributes, keys, cardinalities,generalization, specialization etc.) Convert the ER diagram into relational tables and normalize the Relational data model.
5	Create a database with following schemas Borrower (Rollin, Name, Date of Issue, Name of Book, Status) & Fine (Roll_no, Date, Amt)
6	**Cursors** Write a block in PL/SQL to print a report which shows that, the employee id, name, hire date, and the incentive amount they achieved according to their working experiences, who joined in the month of current date. Use explicit cursor
7	**Database Trigger** Create a Library database with the schema – Books (AccNo, Title, Author, Publisher, Count). a.Create a table Library_Audit with same fields as of Books and Date and status column b.Create a before trigger to insert records into Librry_Audit table if there is deletion in Books table, insert date of deletion and status as deleted Create a after trigger to insert records into Librry_Audit table if there is updation in Books table , insert date of updation and status as updated.
8	**Database Connectivity:** Write a program to implement menu driven MySQL/Oracle database connectivity with any front end language for Python/Java/PHP to implement Database navigation operations (add, delete, edit etc.)
9	MongoDB Queries Create a collection named movies. Insert 5 records with fields: TITLE, DESCRIPTION, DIRECTED_BY, URL, GENRES, and LIKES. Insert 1 more document in the collection with additional fields: username and comments. 1.Display all the documents whose title is 'inception'. 2.Display all the documents directed by 'Raj' or whose title is 'inception' 3.Display all the documents whose title is 'inception' and directed by 'Raj' 4.Display all the documents whose likes are greater than 10. 5.Display all the documents whose likes are greater than 100 and whose title is either 'inception' or directed by 'Raj' 6.Update the title 'inception' to 'inception overview'. 7.Delete the document titled 'thriller overview'. 8.Display exactly two documents directed by 'Raj'. 9.Display the second document directed by 'Raj'. 10.Display all the movies in sorted fashion. Insert a document using the save() method.
10	MongoDB Aggregation and Indexing Create the collection Movies having the following fields: TITLE, DESCRIPTION, DIRECTOR, URL, GENRES, and LIKES Implement the following Aggregation and Indexing Queries: 1.Find the number of movies directed by "Raj". 2.Find movies which have minimum likes and maximum likes directed by "Raj". 3.Find the average number of likes of the movies directed by "Raj". 4.Find the first and last movie directed by "Raj". 5.Create an index on the director's name. 6.Display the movies directed by "Raj" and check if it uses the index which we have created.

11	**Mini Project:** Form a group of 3 or 4 students and Using the database concepts covered, develop an application with following details: 1.Define a problem statement 2.Follow the Software Development Life cycle and other concepts learnt in Software Engineering Course throughout the implementation. 3.Develop application considering: 4.Front End: Java/Perl/PHP/Python/Ruby/.net/any other language 5.Front End: Java/Perl/PHP/Python/Ruby/.net/any other language 6.Test and validate applications using Manual/Automation testing.

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
MDM-221-CSD: Digital Electronics		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	02	**CCE** : 30 Marks **End-Semester**: 70 Marks

Prerequisite Courses, if any :

1. Basic Electrical and Electronics Engineering.

Course Objectives: The course aims to introduce engineering students to the fundamentals of Digital electronics technology, enhance problem-solving abilities, and provide a strong foundation for careers in computing, automation, and embedded systems.

1. To understand Procedure of Logic Minimization.
2. To study Combinational Circuits.
3. To study Sequential Circuits.
4. To learn Moore and Mealy Machine.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Illustrate acquired knowledge to Logic Minimization Problem.
- CO2: Construct Combinational Circuits for given Specification.
- CO3: Design Combinational Circuits for given Specification
- CO4: Construct Sequential Circuits for given Specification.
- CO5: Understand operations of shift registers and Construct Sequential Circuits using Moore and Mealy Machine.

Course Contents

Unit I - Logic Minimization (06 Hours)

Boolean Function Representation: Sum of Product (SOP) and Product of Sum (POS) form of Boolean Expression, Standard SOP and POS form.
Minimization Technique: Karnaugh map (K-map) representation of Logical Function, Simplification of Logical Function using K-map, Minimization of SOP forms, Minimization of POS forms, Implementation of Digital Circuits using Universal gates.

Unit II - Combinational Logic Design - I (06 Hours)

Adder, Subtractor, Look ahead Carry Adder, BCD Adder, Design of Full Adder, Subtractor and BCD Adder
Codes: Binary Code, BCD Code (Binary-Coded Decimal), Excess-3 Code, Gray Code, Alphanumeric Code, Error Detecting and Correcting Codes, Code Converters

Unit III - Combinational Logic Design - II (06 Hours)

Multiplexer, Multiplexer IC 74151, Multiplexer Tree, Demultiplexer, Demultiplexer Tree, Comparator, Encoder, Priority Encoder, Decoder, Combinational Circuit Design using Mux, DeMux
Case Study (Any one): IC 74181 (ALU), BCD to 7-Segment Display Controller, Calendar Subsystem

Unit IV - Sequential Circuit Design-1 (06Hours)

Flip Flop: 1-bit memory cell, Clocked S-R Flip Flop, J-K Flip Flop, race around condition, M/S J-K Flip Flop, D and T Flip Flop, Excitation Table, Flip-Flop Conversion.
Counter: Asynchronous and Synchronous Counters, Design of Asynchronous counter, Modulus Asynchronous Counters, IC 7490, Design of Synchronous Counter, Modulus Synchronous Counter, Case Study: Security Monitoring System.

Unit V -Sequential Circuit Design-2 (06 Hours)

Shift Register: SISO (Serial-In, Serial-Out), SIPO (Serial-In, Parallel-Out), PIPO (Parallel-In, Parallel-Out), PISO (Parallel-In, Serial-Out), Bidirectional Shift Register, Universal Shift Register, Ring and Twisted Ring/Johnson Counter.
Moore / Mealy Machine: Representation Techniques, State Diagrams, State Tables, State Reduction, State Assignment, Implementation using Flip-Flops, Design of Sequence Generator and Detector.

Learning Resources

- **Text Books:**

1. Modern Digital Electronics by R.P.Jain, 4th Edition, ISBN 978-0-07-06691-16 Tata McGraw Hill
2. Digital Logic and Computer Design by Moris Mano, Pearson , ISBN 978-93-325-4252-5

- **Reference Books:**

1. John F. Wakerly, "Digital Design: Principles and Practices," Pearson.
2. Mark Bach, “Complete Digital Design”, Tata MCGraw Hill, 2005.
3. Charles H. Roth Jr., "Fundamentals of Logic Design," Cengage Learning.

- **e-Books:**

1. https://link.springer.com/book/10.1007/978-3-030-36196-9
2. https://www.mheducation.co.uk/ebook-fundamentals-of-digital-logic-9780077144227-emea

- **MOOC / NPTEL/YouTube Links: -**

1. Digital Circuits, by Prof.SantanuChattopadhyay , IIT Kharagpur https://swayam.gov.in/nd1_noc19_ee5
2. Digital Circuits and Systems ,Prof. S. Srinivasan , IIT Madras https://nptel.ac.in/courses/117/106/117
3. Microprocessors and Interfacing, by Prof Shaikh Rafi Ahamed, IIT Guwahati.https://onlinecourses.npte
4. VLSI Technology”, by Dr. Nandita Dasgupta, IIT Madras https://nptel.ac.in/courses/117106093

- **YouTube/Video Links:**

1. https://www.youtube.com/watch?v=CL3ups78jrs
2. https://www.youtube.com/watch?v=ibQBb5yEDlQ

Savitribai Phule Pune University Second Year Engineering - Computer Science and Design(2024 Pattern)		
EEM-231-CSD: Entrepreneurship Development		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical : 2 Hours/Week Tutorials : 1 Hour/Week	02	Term Work : 25 Marks

Companion Course : Entrepreneurship Development Laboratory

Course Objectives: The course aims to:

- To develop an entrepreneurial mindset in students.
- To introduce students to the process of idea generation, business planning, and implementation.
- To provide students with practical knowledge of how to start, grow, and manage a business.
- To analyze and learn from real-world case studies of successful and failed entrepreneurial ventures.
- To help students understand the challenges and risks faced by entrepreneurs, particularly in the tech industry

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Understand the key elements of entrepreneurship.
- CO2: Be able to identify business opportunities and validate business ideas.
- CO3: Learn to create a detailed business plan.
- CO4: Understand start up finance and marketing strategies.
- CO5: Learn from case studies to avoid common pitfalls and leverage successful strategies

Course Contents

Unit I - Introduction to Entrepreneurship (03 Hours)

Introduction, Concept of Entrepreneur, Entrepreneurship and Enterprise, Definition of Entrepreneurship, Objectives of Entrepreneurship Development, Phases of Entrepreneurship Development, Role of Entrepreneurship, The Entrepreneurial Mindset, Characteristics of Entrepreneurship, Traits of Entrepreneurship, Introduction to Entrepreneurship Skills

Case Study: The Journey of Steve Jobs: The Rise of Apple Inc.

Unit II -Entrepreneurship Development Skills (03 Hours)

Meaning of Entrepreneurship skill, Types of Entrepreneurship Skills: Business management skills, Teamwork and leadership skills, Communication and listening, Customer service skills, Financial skills, Analytical and problem-solving skills, Critical thinking skills, Strategic thinking and planning skills, Technical skills, Time management and organizational skills, Branding, marketing and networking skills , How to improve entrepreneurial skills, Entrepreneurial skills in the workplace, Entrepreneurial Imagination And Creativity

Case Study: The Entrepreneurship Development Journey of Sara Blakely (Spanx)

Unit III - Creating and Starting the Venture (03 Hours)

Sources of new Ideas, Methods of generating ideas, creating problem solving, product planning and development process

Case Study Airbnb emerged from the founders' personal need to make extra income by renting out space.

Unit IV - The Business Plan (03 Hours)

Nature and scope of Business plan, Writing Business Plan, Evaluating Business plans, Using and implementing business plans. Marketing plan, financial plan and the organizational plan, Launching formalities.

Case Study: Tesla's Disruptive Model: Tesla's business plan

Unit V - Financing and Managing the new venture (03 Hours)

Sources of capital, Record keeping, recruitment, motivating and leading teams, financial controls. Marketing and sales controls. E-commerce and Entrepreneurship, Internet advertising

Case Study: Amazon's Rise from Garage to Global: Jeff Bezos started Amazon with a clear business plan, sourcing capital from personal savings and angel investors.

Learning Resources

Text Books:

1. Donald F. Kuratko, "Entrepreneurship: Theory, Process, and Practice", South-Western Publ., ISBN-13 978-0324258264
2. Vasant Desai, K. Kaur, "Entrepreneurship: Development and Management", Himalaya Publishing House, ISBN-13 978-9350978757
3. Bruce R. Barringer and R. Duane Ireland, "Entrepreneurship: Successfully Launching New Ventures", 6th Ed. Pearson Education, ISBN-13 978-9353066499

Reference Books:

1. Eric Ries, "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses"
2. Robert Hisrich and Michael Peters, "Entrepreneurship", 9th Ed., Tata Mc Graw-Hill, ISBN-13 978-0078029196
3. Marc J Dollinger, "Entrepreneurship - Strategies and Resources", Irwin Professional Publishing, ISBN-13 978-0256119800
4. Vasant Desai, "Small Scale Industries & Entrepreneurship", Himalaya Publishing House, ISBN-13 978-9350973769

MOOC / NPTEL/YouTube Links: -

1. https://onlinecourses.nptel.ac.in/noc20_mg35/preview
2. https://onlinecourses.nptel.ac.in/noc25_ge11/preview
3. https://onlinecourses.swayam2.ac.in/ntr24_ed51/preview
4. https://onlinecourses.swayam2.ac.in/cec20_mg19/preview

YouTube/Video Links

1. https://www.youtube.com/watch?v=rA4uKIy5gO0&list=PLsh2FvSr3n7fQlIDbfKutmSL26TsWitGQ&i
2. https://youtu.be/cNUiBt5_ZaM?si=NGKomhNiPdTY2I7p
3. https://youtu.be/mCcm9pRXW9M?si=KvjS4FIGf2EwBA_I
4. https://youtu.be/XSNXtAGLA5k?si=Fvi1bKJTeGEtq7c2
5. https://youtu.be/mGCCaIfG9us?si=Bc6MnY1Q2EyZsohp
6. https://youtu.be/RKhcqYh4T0c?si=MnkGvqVfd8X3MlaD

List of Assignments

No	Title	Objective	Description
1	Entrepreneurial Mindset Reflection	To encourage students to explore their personal views on entrepreneurship and recognize the key characteristics of an entrepreneurial mindset by studying the journey of a real-world entrepreneur.	**Write a reflective essay** (500–600 words) based on the following: • Explain what entrepreneurship means to you personally. • Identify an entrepreneur (Indian or global) whom you admire and explain the reasons for your admiration. • Highlight specific mindset traits (e.g., risk-taking, resilience, innovation, adaptability) that contributed to this entrepreneur's success. • Reflect on how these traits align with your own strengths or indicate areas you wish to develop.
2	Idea Generation Challenge	To foster creativity, structured brainstorming, and the ability to identify potential business opportunities based on real-world problems.	**Generate 10 Business Ideas** Use any structured brainstorming technique Ideas can be tech-based, social impact, service-based, or product-based 2. Select One Idea- Choose the most promising idea from your list 3. Write a 1-page Concept Summary, include the following: • Problem Identified: Describe the specific problem or pain point your idea addresses. • Solution Overview: Briefly describe your business idea. • Target Audience: Identify the group of people or organizations that would benefit. • Market Potential: Discuss the viability and scalability of the idea.
3	Business Model & Customer Validation	To help students develop a clear, structured business model and test its assumptions through customer conversations. The goal is to learn how to validate ideas through real-world feedback and refine the business concept accordingly.	**Part A: Business Model Canvas** 1. Choose a business idea (from Assignment 2 or a new one). 2. Create a Business Model Canvas with all 9 key blocks: o Customer Segments o Value Propositions o Channels o Customer Relationships o Revenue Streams o Key Resources o Key Activities o Key Partnerships o Cost Structure 3. Present the BMC in visual or tabular format.

			Part B: Customer Interviews & Insights 1. Identify 2–3 potential customers from your target segment. 2. Conduct brief interviews (5–10 minutes each) to gather insights on: o Their pain points o Their reaction to your proposed solution o Willingness to pay or use your product/service 3. Summarize findings in a 1–1.5 page report that includes: o Key customer quotes or paraphrased insights o A revised Value Proposition or Customer Segment block (if needed) o A short reflection: key learnings and potential changes to your idea
4	Business Launch Plan – Marketing & Financial Snapshot	To develop a practical understanding of how marketing stratey and financial planning go hand-in-hand in launching a startup. Students will define a basic marketing campaign and align it with estimated costs, pricing, and projected revenue.	You are preparing to launch your business idea. Prepare a combined Marketing and Financial Snapshot including the following **Part A: Marketing Campaign Plan** • Define your target market by identifying primary customers. • Design a mini-campaign using one or more of the following channels: Social media (e.g., Instagram, LinkedIn) Print/digital flyers Email marketing • Describe the campaign content, including the message or offer to be promoted. • Optionally, create 1–2 sample marketing materials. • Write a 300-word explanation outlining your marketing strategy and expected impact.
			Part B: Financial Snapshot 1. Startup Costs – Estimate your initial costs (fixed + variable) 2. Pricing Strategy – State your pricing model and justification 3. Break-even Analysis – Basic cost vs. sales estimate 4. 6-Month Revenue Projection – Expected sales and income 5. Format: Use a simple table or spreadsheet (optional)

5	Elevator Pitch Video	To help students develop confidence and clarity in presenting their business idea in a short, compelling format. The exercise simulates real-world investor or networking scenarios where entrepreneurs must grab attention quickly.	**Prepare a 90-second elevator pitch for your business idea** (the same or refined idea used in earlier assignments). Your pitch should cover the following elements: o The Problem – Problem Identification o The Solution – Description of your product/service. o Value Proposition – The unique value proposition. o Target Audience – Audience for your idea. o Call to Action – E.g. request for support, funding, feedback, etc. **Deliver Your Pitch:** o Record a video and submit it with written version of your pitch. o Ensure clear speech, confident body language (for video), and persuasive tone. **Reflection (Short Write-up):** o Share what you learned about communicating your idea o Describe challenges or rewards you experienced in the process

Savitribai Phule Pune University Second Year Engineering - Computer Science and Design(2024 Pattern)		
VEC-232-CSD: Universal Human Values and Professional Ethics		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	02	**CCE** : 15 Marks **End-Semester**: 35 Marks

Prerequisite Courses, if any :
UHV-1 of Student Induction Program (SIP) (desirable)
Course Objectives: The course aims to:

1. To help the students develop a holistic, humane world-vision, and appreciate the essential complementarity between values and skills to ensure mutual happiness and prosperity
2. To elaborate on 'Self-exploration' as the process for Value Education
3. To facilitate the understanding of harmony at various levels starting from self and going towards family and society.
4. To elaborate on the salient aspects of harmony in nature and the entire existence
5. To explain how the Right understanding forms the basis of Universal human values and definitiveness of Ethical human conduct.
6. To provide the vision for a holistic way of living and facilitate transition from chaotic life to an orderly life.

Course Outcomes: Upon successful completion of this course, students will be able to:

1. Recognize the concept of self-exploration as the process of value education and see they have the potential to explore on their own right.
2. Explore the human being as the coexistence of self and body to see their real needs / basic aspirations clearly.
3. Explain relationship between one self and the other self as the essential part of relationship and harmony in the family.
4. Interpret the interconnectedness, harmony and mutual fulfilment inherent in the nature and the entire existence.
5. Draw ethical conclusions in the light of Right understanding facilitating the development of holistic technologies production systems and management models.

Course Contents

Unit I - Introduction to Value Education (07 Hours)

(i) Understanding Value Education
(ii) Self-exploration as the Process for Value Education
(iii) Continuous Happiness and Prosperity - the Basic Human Aspirations and their Fulfilment
(iv) Right Understanding, Relationship and Physical Facility
(v) Happiness and Prosperity - Current Scenario
(vi) Method to Fulfil the Basic Human Aspirations

Unit II - Harmony in the Human Being (07 Hours)

(i) Understanding Human being as the Co-existence of the Self and the Body
(ii) Distinguishing between the Needs of the Self and the Body

(iii) The Body as an Instrument of the Self
(iv) Understanding Harmony in the Self
(v) Harmony of the Self with the Body
(vi) Programme to Ensure self-regulation and Health

Unit III -Harmony in the Family and Society (08Hours)

(i) Harmony in the Family - the Basic Unit of Human Interaction "Trust' - the Foundational Value in Relationship
(ii) 'Respect' - as the Right Evaluation
(iii) Values in Human-to-Human Relationship
(iv) Understanding Harmony in the Society
(v) Vision for the Universal Human Order

Unit IV -Harmony in the Nature (Existence) (08 Hours)

(i) Understanding Harmony in the Nature
(ii) Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature
(iii) Realizing Existence as Co-existence at All Levels
(iv) The Holistic Perception of Harmony in Existence

Learning Resources

Text Books:

1. A Foundation Course in Human Values and Professional Ethics, RR Gaur, R Asthana, GP Bagaria, 3rd revised edition, UHV Publications, 2023, ISBN: 978-81-957703-7-3 (Printed Copy), 978-81-957703-6-6 (e-book)
2. Teachers' Manual for A Foundation Course in Human Values and Professional Ethics, RR Gaur, R Asthana, GP Bagaria, 3rd revised edition, UHV Publications, 2023, ISBN: 978-81-957703-5-9 (Printed Copy), 978-81-957703-0-4 (e-Book)

Reference Books:

1. P. L. Dhar, R. R. Gaur, 1990, Science and Humanism, Commonwealth Publishers.
2. A. Nagaraj, 1999, Jeevan Vidya: Ek Parichaya, Jeevan Vidya Prakashan, Amarkantak
3. B. P. Banerjee, 2005, Foundations of Ethics and Management, Excel Books.
4. A. N. Tripathy, 2003, Human Values, New Age International Publishers.
5. E. G. Seebauer & Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers , Oxford University Press
6. B. L. Bajpai, 2004, Indian Ethos and Modern Management, New Royal Book Co., Lucknow. Reprinted 2008.
7. M. Govindrajran, S Natrajan & V.S. Senthil Kumar, Engineering Ethics and Human Values, Eastern Economy Edition, Prentice Hall of India Ltd.
8. M. K. Gandhi, "The Story of my Experiments with Truth", Discovery Publisher

MOOC / NPTEL/YouTube Links: -

1. Swayam Course on "Understanding Human Being Nature and Existence Comprehensively" by Dr. Kumar Sambhav, Director, UP Institute of Design (UPID), Noida. https://onlinecourses.swayam2.ac

2. NPTEL Course on “Exploring Human Values: Visions of Happiness and Perfect Society” by Prof. A. K. Sharma, Department of Humanities and Social Sciences, IIT Kanpur. https://nptel.ac.in/courses/1

E-Resources: -

1. https://fdp-si.aicte-india.org/download.php#1/
2. https://madhyasth-darshan.info/postulations/knowledge/knowledge-of-humane-conduct/
3. https://www.youtube.com/channel/UCQxWr5QB_eZUnwxSwxXEkQw

Savitribai Phule Pune University Second Year Engineering - Computer Science and Design(2024 Pattern)		
CEP-241- CSD: : Community Engagement Project		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical : 04 Hours/Week	02	**Term Work :** 25 Marks **Oral:**25 Marks

Prerequisite Courses:

- Basic understanding of social and ethical responsibilities
- Teamwork and communication skills acquired in prior coursework or group activities
- Familiarity with problem-solving methodologies and project planning
- Conversation in local language

Companion Course :

1. CEP is an experiential learning approach that combines education, learning, community development, and meaningful community service.
2. Project involves students in community development and service activities and applies the experience to personal and academic development.
3. The targeted contribution of college students to the village/local development will benefit the community.
4. The college has an opportunity to help students become more socially conscious and responsible while simultaneously becoming a socially conscious organization.

Course Objectives: The course aims to:

1. Establish a mutually beneficial relationship between the college and the community
2. Opportunities to engage with their local community, fostering empathy, teamwork, and problem-solving skills while contributing positively to their surroundings.
3. An understanding of the challenges faced by the local community and the role of engineering in addressing those challenges.
4. The ability to apply technical knowledge and skills to design solutions or interventions that create a positive impact on the community.
5. The skills to evaluate and critically analyze the outcomes of their engagement activities, deriving actionable insights for sustainable impact

Course Outcomes: Upon successful completion of this course, students will be able to:

1. CO1 - **Identify** and **Analyze** local community needs and challenges by engaging with stakeholders and evaluating real-world problems.
2. CO2- **Design** and **Implement** practical, creative, and context-specific solutions using engineering principles to address community issues.
3. CO3 - **Reflect** and **Evaluate** the effectiveness of their interventions and articulate lessons learned through reports and presentations.

Course Contents

Implementation

- A group of 3 to 4 students or a single student could be assigned for a particular habitation or village or municipal ward, as far as possible, in the near vicinity of their place of stay/college premise.
- Each group is allotted to a faculty member of the department as a mentor.
- The group of students will be associated with a government official / village authorities /NGOs etc. concerned, allotted by the district administration, during the duration of the project.
- The Community Engagement Project should be different from the regular programmes of NSS/NCC/Gr Club/Hobby Clubs, Special Interests Groups etc
- An activity book has to be maintained by each of the students to record the activities undertaken/involved and will be countersigned by the concerned mentor/HoD.
- Project report shall be submitted by each student/group of students.
- An internal evaluation shall also be conducted by a committee constituted by the HoD. Evaluation to be done based on the active participation of the student and marks could be awarded by the mentor/HoD.
- Students groups can conduct an awareness programme on Health and Hygiene or in Organic Farming or in Fisheries or in advocating prohibition of liquor or about renewable energy, e-waste management or any other activity in an area of their studies and as per his/her aptitude.

Suggestive list of topics under Community Engagement Project

The below lists are not exhaustive and open for HoD's or mentors to add, delete or modify. It is expected that the focus should be on specific local issues in their nearby areas.

The students are expected to carry out these projects with involvement, commitment, responsibility and accountability. The mentors of a student/group of students shall

- Use and/or miss-use of cell phones
- Career orientation of youth
- Water facilities and drinking water availability
- Health and hygiene of the school going students, home makers and old personals
- Health intervention and awareness programmes
- Horticulture
- Herbal and Nutrition
- Traditional and Modern health care methods
- Food habits
- Air /Sound /Water pollution
- Plantation and Soil protection
- Renewable energy and Solar Systems
- Yoga awareness and practice
- Health care awareness programmes and their impact

- Organic farming
- Food adulteration
- Incidence of Diabetes and other chronic diseases
- Blood groups and blood levels
- Chemicals in daily life
- Music and dance
- Women education and empowerment

Project Scope

- Conduct workshops or awareness drives on topics like digital literacy, environmental sustainability, mental health, or career planning for local stakeholders.
- Develop a simple prototype or solution that addresses a real-world problem (e.g., a water-saving device, simple mobile apps, or tools for community use).
- Organize clean-up drives, tree plantations, recycling campaigns, or energy conservation initiatives.
- Promote health through awareness programs on hygiene, nutrition, and exercise.
- Teach basic computer or technical skills to students, staff, or the community

Proposal Submission

CEP Group should Submit a two-page project proposal, preferably prior to the term commencement outlining the following:-

- Title of the project
- Aim, Objective and expected outcome
- Plan of execution (timeline and activities).
- Place of the CEP and involvement of any local authority, NGP
- Required resources (if any).
- Get approval from the designated faculty mentor.

Learning Resources

Text Books:

1. Waterman, A. Service-Learning: A Guide to Planning, Implementing, and Assessing Student Projects. Routledge, 1997.
2. Beckman, M., and Long, J. F. Community-Based Research: Teaching for Community Impact. Stylus Publishing, 2016.
3. Design Thinking for Social Innovation. IDEO Press, 2015.

4. Dostilio, L. D., et al. The Community Engagement Professional's Guidebook: A Companion to The Community Engagement Professional in Higher Education. Stylus Publishing, 2017

- **MOOC / NPTEL/YouTube Links:**

1. NPTEL course: Ecology and Society https://onlinecourses.nptel.ac.in/noc20_hs77/preview

Web Links: -

1. UNESCO: Education for Sustainable Development https://www.unesco.org
2. EPICS (Engineering Projects in Community Service) https://engineering.purdue.edu/EPICS
3. Ashoka: Innovators for the Public https://www.ashoka.org
4. Design for Change https://www.dfcworld.com

Savitribai Phule Pune University, Pune

Maharashtra, India

SE - Computer Science and Design

2024 Pattern

Semester IV

With effect from Academic Year 2025-26

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-251-CSD :Advance Data Structures		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Programming and Problem Solving , Data Structure and Algorithms
Companion Courses: Advance Data Structures Laboratory
Course Objectives: The course aims to:

1. To develop a logic for graphical modeling of the real life problems.
2. To suggest appropriate data structure and algorithm for graphical solutions of the problems.
3. To understand advanced data structures to solve complex problems in various domains.
4. To operate on the various structured data
5. To build the logic to use appropriate data structure in logical and computational solutions.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Identify and articulate the complexity goals and benefits of a good hashing scheme for real- world applications.
- CO2: Apply non-linear data structures for solving problems of various domains.
- CO3: Design and specify the operations of a nonlinear-based abstract data type and implement the min a high-level programming language.
- CO4: Analyze the algorithmic solutions for resource requirements and optimization
- CO5: Use efficient indexing methods and multiway search techniques to store and maintain data.

Course Contents

Unit I - Hashing (09 Hours)

Hash Table- Concepts-hash table, hash function, basic operations, bucket, collision, probe, synonym, overflow, open hashing, closed hashing, perfect hash function, load density, full table, load factor, re-hashing, issues in hashing, hash functions- properties of good hash function, division, multiplication, extraction, mid-square, folding and universal, Collision resolution strategies- open addressing and chaining, Hash table overflow- open addressing and chaining, extendible hashing, closed addressing and separate chaining. Skip List- representation, searching and operations- insertion, removal.

Unit II - Trees (09 Hours)

Tree- basic terminology, General tree and its representation, representation using sequential and linked organization, Binary tree- properties, converting tree to binary tree, binary tree traversals (recursive and non-recursive)- inorder, preorder, post order, depth first and breadth first, Operations on binary tree. Huffman Tree (Concept and Use), Binary Search Tree (BST), BST operations, Threaded binary search tree- concepts,

Unit III - Graphs (09Hours)

Basic Concepts, Storage representation, Adjacency matrix, adjacency list, adjacency multi list, inverse adjacency list. Traversals-depth first and breadth first, Minimum spanning Tree, Greedy algorithms

for computing minimum spanning tree- Prims and Kruskal Algorithms, Dikjtra's Single source shortest path, All pairs shortest paths- Flyod-Warshall Algorithm, Topological sort ordering.

Unit IV - Search Trees (09 Hours)

Symbol Table-Representation of Symbol Tables- Static tree table and Dynamic tree table, Weight balanced tree - Optimal Binary Search Tree (OBST), OBST as an example of Dynamic Programming, Height Balanced Tree- AVL tree. Red-Black Tree, AA tree, K-dimensional tree, Splay Tree

Unit V - Indexing and Multiway Trees (09 Hours)

Indexing and Multiway Trees- Indexing, indexing techniques-primary, secondary, dense, sparse, Multiway search trees, B-Tree- insertion, deletion, B+Tree - insertion, deletion, use of B+ tree in Indexing, Trie Tree.

Learning Resources

Text Books:

1. Horowitz, Sahani, Dinesh Mehata, "Fundamentals of Data Structures in C++, Galgotia Publisher, ISBN: 8175152788, 9788175152786.
2. Peter Brass, "Advanced Data Structures", Cambridge University Press, ISBN: 978-1-10743982-5
3. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, "Data Structures and Algorithms in Python", Wiley Publication, ISBN:978-1-118-29027-9.

Reference Books:

1. A. Aho, J. Hopcroft, J. Ulman, "Data Structures and Algorithms", Pearson Education, 1998, ISBN-0-201-43578-0.
2. Michael J Folk, "File Structures an Object Oriented Approach with C++", Pearson Education, ISBN: 81-7758-373-5.
3. Sartaj Sahani, "Data Structures, Algorithms and Applications in C++", Second Edition, University Press, ISBN:81-7371522 X.

MOOC / NPTEL/YouTube Links: -

1. https://nptel.ac.in/courses/106/102/106102064/
2. https://nptel.ac.in/courses/106/105/106105085
3. https://nptel.ac.in/courses/106/106/106106127
4. https://nptel.ac.in/courses/106/102/106102064/
5. https://nptel.ac.in/courses/106/105/106105085
6. https://nptel.ac.in/courses/106/106/106106127

e-Books:-

1.https://www.ebooks.com/en-us/book/95777110/Python-data-structures-and-algorithms/benjamin-baka/

2.https://www.ebookphp.com/advanced-data-structures-epub-pdf/

3.https://www.ebookphp.com/data-structures-and-algorithms-professional-edition-beginners-guide-epub-pdf/

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-252-CSD: System Programming and Operating System		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	02	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses : Programming and Problem solving , Data Structures and Algorithms

Course Objectives: To provide students with a foundational understanding of computer evolution, memory management, the 8086 microprocessor, memory organization, interrupts, and parallel organization in computer systems.

1. To get acquainted with the basics of System Programming
2. To be familiar with the format of object modules, the functions of compiler, linking, and loading.
3. To comprehend the structures and functions of Operating Systems and process management.
4. To learn and understand memory management of Operating System
5. To learn and understand open source Operating System .

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Understand the fundamentals of systems programming.
- CO2: Study of Compilers, linkers and loaders.
- CO3: Analyze the performance of process scheduling algorithms
- CO4: Understand and apply memory management policies
- CO5: Study Linux Operating System .

Course Contents

Unit I - Introduction (06 Hours)

Introduction to Systems Programming, Need of systems programming, Software Hierarchy, Types of software: system software and application software.
Evolution of components of systems programming: Text Editors, Assembler, Macros, Compiler, Interpreter, Loader, Linker, Debugger, Device Drivers, Operating System.

Case Study: Evolution (Brief History) of Microprocessors

Unit II - Compilers, Linkers and Loaders (06 Hours)

Introduction to Compilers: Phases of Compiler with one example, Comparison of compiler and Interpreter.
Introduction to Linkers: Definition, purpose, and importance of linking in system programming.
Linker Types: Static linker, dynamic linker, and load-time linker.
Introduction, Loader schemes: Compile and Go, General Loader Scheme, Absolute Loaders.

Unit III - Operating System (06 Hours)

Introduction: Evolution of OS, Operating System Services, Functions of Operating System.
Process Management: Process, Process States: 5 and 7 state model, process control block
Process Scheduling: Uni-processor Scheduling, Scheduling: Preemptive, Non-preemptive, Long-term, Medium-term, Short term scheduling.
Scheduling Algorithms: FCFS, SJF, RR, and Priority.

Unit IV - Memory Management (06 Hours)

Introduction: Memory Management concepts, Memory Management requirements.
Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Paging, Segmentation, Address translation.
Placement Strategies: First Fit, Best Fit, Next Fit and Worst Fit.
Page Replacement Policies: First In First Out (FIFO), Last Recently Used (LRU), Optimal.

Unit V - Introduction to Linux (06 Hours)

Overview of Linux: History, features, Structure, Basic commands and advantages of Linux
Linux file system: File system hierarchy, file types, and permissions
Linux Processes and Job Control: Process management: Creating, listing, and managing processes

Learning Resources

Text Books:

1. John Donovan, “System Programming”, McGraw Hill, ISBN 978-0--07-460482-3.
2. Dhamdhere D., "Systems Programming and Operating Systems", McGraw Hill, ISBN 0 - 07 - 463579 – 4
3. Silberschatz, Galvin, Gagne, "Operating System Principles", 9th Edition, Wiley, ISBN 978- 1-118-06333-0

Reference Books:

1. Leland Beck, “System Software: An Introduction to systems programming”, Pearson, ISBN-13978-0201423006
2. John R. Levine, Tony Mason, Doug Brown, “Lex & Yacc”, 1st Ed., O’REILLY, ISBN 81-7366-062-X.
3. Alfred V. Aho, Ravi Sethi, Reffrey D. Ullman, “Compilers Principles, Techniques, and Tools”, Addison Wesley, ISBN 981-235-885-4
4. Andrew Tannenbaum “Modern Operating Systems”, 5th Ed., ISBN 9780137618880
5. Richard Petersen, “Linux: The Complete Reference”, 6th Ed., McGraw-Hill Osborne Media, ISBN-13978-0071492478

Savitribai Phule Pune University Second Year of Computer Science Design (2024 Course)		
PCC-253-CSD: Computer Graphics		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 03 Hours/Week	03	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses :Basic Mathematics

Companion Course : Computer Graphics Laboratory

Course Objectives: The course aims to:

1. Remembering: To acquaint the learner with the basic concepts of Computer Graphics.
2. Understanding: To learn the various algorithms for generating and rendering graphical figures.
3. Applying: To get an idea of modeling transformations, clipping and polygon filing.
4. Understanding: To understand various methods of projections and color models, and techniques of animation and gaming.
5. Creating: To create interactive programs using OpenGL

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Identify the basic terminologies of Computer Graphics and apply mathematics for elementary graphic operations for scan conversion of line and circle.
- CO2: Illustrate the concepts of windowing and clipping and apply various algorithms to fill and clip polygons.
- CO3: Apply the basic modeling transformations in both two and three dimensions.
- CO4: Understand the concepts of color models, hidden surface elimination and apply mathematics to generate curves and fractals.
- CO5: Understand basics of computer animation, gaming and parallelism and generate interactive graphics using OpenGL.

Course Contents

Unit I - Introduction to Computer Graphics and Scan Conversion (09 Hours)

Introduction to computer graphics primitives: Pixel, Resolution, Aspect ratio, Frame buffer, Computer graphics Applications.

Display Devices: Random scan display, Raster scan display, OLED, Micro-OLED, Active Matrix OLED (AMOLED), Quantum Dot (QLED)

Scan conversion: Line drawing algorithms: Digital Differential Analyzer (DDA), Bresenham; Circle drawing algorithms: DDA, Bresenham, and Midpoint; Introduction to aliasing and anti-aliasing.

#Exemplar/Case studies:

1. Real-Time Map Rendering in GPS Navigation Systems using Line and Circle Drawing Algorithms

Unit II -Polygons and Clipping (09 Hours)

Polygons: Types: Convex, Concave, and Complex. Edge table, Vertex inside-outside tests: Odd-even rule, Winding number rule.

Polygon Filling Algorithms: Seed Fill, Flood Fill (4-connected and 8-connected), Scan Line Polygon Fill, Filling with patterns.

2D Viewing: Viewing Pipeline, Window-to-Viewport Coordinate Transformation.

2-D Clipping: Point Clipping, Line Clipping, Cohen-Sutherland Line Clipping Algorithm.

Polygon Clipping: Sutherland-Hodgeman Polygon Clipping Algorithm, Weiler-Atherton Polygon Clipping Algorithm, Interior and Exterior Clipping, Text clipping.

#Exemplar/Case Studies - 3D pipeline / polygonal modelling and applications

Unit III - 2D and 3D Transformations (09 Hours)

2D Transformations: Introduction, Basic Transformations: Translation, Rotation, Scaling, Shear; Matrix Representation and Homogeneous Coordinate System, Composition of Transformations; Rotation, scaling, and shearing with respect to any arbitrary fixed point (other than origin);

3D Transformations: Translation, Rotation, Scaling, Shear; 3D Rotation about any arbitrary axis.

Types of Projections: Parallel (Oblique: Cavalier, Cabinet and orthographic: isometric, diametric, trimetric) and Perspective (Vanishing Points: 1 point, 2 point and 3 point)

#Exemplar/Case Studies - 1Affine Transformations Vlab (Vlab link: https://cse19-iiith.vlabs.ac.in/exp/affi transformation/theory.html)

Image augmentation in Deep learning

Unit IV - Hidden Surface Removal Methods, Curves and Fractals (09 Hours)

Color Models: Properties of Light, Standard primaries, XYZ Color Model, CIE Chromaticity Diagram, Intuitive color concepts, RGB, CMY and HSV Color models, Color selection applications.

Hidden-Surface Removal Methods: Types of Methods, Back Face Elimination, Depth (Z) Buffer Algorithm, Depth Sorting (Painter's) Algorithm, Area-Subdivision (Warnock's) Algorithm.

Curves: Introduction, Curve generation, Interpolation, Blending functions, B-Spline curve, Bezier curve.

Fractals: Classification of fractals, Fractal dimension, Geometric construction of deterministic self-similar fractals: Snowflake, Koch curve, Hilbert curve, Applications.

#Exemplar/Case study :Fractal generation software

Unit V - Applications (09 Hours)

Computer Animation: Introduction, conventional and computer-based animation, design of animation sequences, key-frame, morphing.

Gaming: Introduction, Gaming Platform: NVidia, advances in gaming.

GPU and GPU Programming: Need for parallelism, multicore, streamlining multiprocessors, GPU organization, vertex shader, fragment shader.

OpenGL: Special purpose packages vs. general programming packages, Graphics software standards, History, OpenGL architecture, Programming with OpenGL, Basic OpenGL syntax, GLUT library functions, primitives and attributes, simple modelling and rendering of two and three dimensional objects.

#Exemplar/Case Studies - 1) Relation between computer vision and computer graphics

Learning Resources

Text Books:

1. S. Harrington, "Computer Graphics", 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07–100472 – 6.
2. Donald D. Hearn and Baker, "Computer Graphics with OpenGL", 4th Edition, ISBN-13:9780136053583
3. D. Rogers, J. Adams, "Mathematical Elements for Computer Graphics", 2nd Edition, Tata McGraw Hill Publication, 2002, ISBN 0 – 07 – 048677 – 8.

Reference Books:

1. Dr. Samit Bhattacharya, "Computer Graphics", Oxford University Press, ISBN-13: 978-0-19-809619-1; ISBN-10: 0-19-809619-4.
2. D. Rogers, "Procedural Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.
3. Steve Marschner, Peter Shirley et al, "Fundamentals of Computer Graphics", 4th Edition, CRC Press, ISBN-13: 978-1-4822-2941-7.
4. J. Foley, V. Dam, S. Feiner, J. Hughes, "Computer Graphics Principles and Practice", 2nd Edition, Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.

E-Book

1. https://www.iitg.ac.in/samit/Computer%20Graphics.pdf
2. https://open.umn.edu/opentextbooks/textbooks/introduction-to-computer-graphics
3. http://www2.cs.uidaho.edu/~jeffery/courses/324/lecture.html

MOOC/NPTEL/SWAYAM Course Links:

1. https://archive.nptel.ac.in/courses/106/103/106103224/
2. https://archive.nptel.ac.in/courses/106/102/106102065/
3. https://nptel.ac.in/courses/106106090

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-254-CSD: Advance Data Structures Laboratory		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week	01	Term Work: 25 Marks Practical: 25 Marks

Prerequisite Courses : -

Course Objectives: The course aims to:

1. To understand practical implementation and usage of nonlinear data structures for solving problems of different domain.
2. To strengthen the ability to identify and apply the suitable data structure for the given real world problems.
3. To analyze advanced data structures including hash table, dictionary, trees, graphs, sorting algorithms and file organization.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1. Understand the ADT/libraries, hash tables and dictionary to design algorithms for a specific problem.
- CO2: Apply and analyze nonlinear data structures to solve real world complex problems.
- CO3: Apply and analyze algorithm design techniques for sorting, multi-way searching.

Course Contents

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be done based on overall performance and Laboratory assignments performance of student. Each Laboratory assignment assessment should be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Practical Examination

TBoth internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student's performance for advanced learning, understanding of the fundamentals, effective and efficient implementation. Consequently, encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. Therefore adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students.

The instructor may set multiple sets of assignments and distribute among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged.In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned. Instructor may also set one assignment or mini-project that is suitable to respective branch beyond the scope of syllabus.

Each student must perform at least 10 assignments and 01 mini project)

Operating System recommended: 64-bit Open source Linux or its derivative

Programming tools recommended: Open Source Programming tool like G++/GCC/java.

Suggested List of Laboratory Experiments/Assignments	
1	Consider telephone book database of N clients. Make use of a hash table implementation to quickly look up client‘s telephone number. Make use of two collision handling techniques and compare them using number of comparisons required to find a set of telephone numbers
2	Implement all the functions of a dictionary (ADT) using hashing and handle collisions using chaining with / without replacement.Data: Set of (key, value) pairs, Keys are mapped to values, Keys must be comparable,Keys must be unique. Standard Operations: Insert(key, value), Find(key), Delete(key)
3	To create ADT that implements the "set" concept.a. Add (new Element) -Place a value into the set, b. Remove (element) Remove the value c. Contains (element) Return true if element is in collection,d. Size () Return number of values in collection Iterator () Return an iterator used to loop over collection, e. Intersection of two sets , f. Union of two sets,g. Difference between two sets,h. Subset
4	A book consists of chapters, chapters consist of sections and sections consist of subsections. Construct a tree and print the nodes. Find the time and space requirements of your method
5	Beginning with an empty binary search tree, construct binary search tree by inserting the values in the order given. After constructing a binary tree -i. Insert new node, , ii. Find number of nodes in longest path from root, iii. Minimum data value found in the tree, iv. Change a tree so that the roles of the left and right pointers are swapped at every node, v. Search a value
6	Construct an expression tree from the given prefix expression eg. +--a*b c/d e f and Traverse it using post order traversal (non recursive) and then delete the entire tree.
7	Implement a file compression algorithm that uses binary tree. Your program should Allow the user to compress and decompress messages containing alphabets using the standard Huffman algorithm for encoding and decoding.
8	Write C++/Java program to Represent a given graph using adjacency matrix/list to perform DFS and using adjacency list to perform BFS. Use the map of the area around the college as the graph. Identify the prominent land marks as nodes and perform DFS and BFS on that.

Suggested List of Laboratory Experiments/Assignments	
9	Write C++/Java program- There are flight paths between cities. If there is a flight between city A and city B then there is an edge between the cities. The cost of the edge can be the time that flight takes to reach city B from A or the amount of fuel used for the journey. Represent this as a graph. The node can be represented by airport name or name of the city. Use adjacency list representation of the graph or use adjacency matrix representation of the graph.
10	Write C++/Java program -You have a business with several offices; you want to lease phone lines to connect them up with each other; and the phone company charges different amounts of money to connect different pairs of cities. You want a set of lines that connects all your offices with a minimum total cost. Solve the problem by suggesting appropriate data structures.
11	SA Dictionary stores keywords and its meanings. Provide facility for adding new keywords, deleting keywords, updating values of any entry. Provide facility to display whole data sorted in ascending/ Descending order. Also find how many maximum comparisons may require for finding any keyword. Use Height balance tree and find the complexity for finding a keyword
12	Implement the Heap/Shell sort algorithm implemented in Java demonstrating heap/shell data structure with modularity of programming language
13	Write a Java program for to Read the marks obtained by second year students. Find out maximum and minimum marks using heap data structure.
Mini-Projects/ Case Study	
14	Design and implement a Contact Manager Application in Java that stores, searches, and manages contact records (like name, phone number, email, etc.). The application should demonstrate how different data structures — such as ArrayList, LinkedList, HashMap, TreeMap, and custom-built data structures — affect the performance of operations like insertion, deletion, searching, and sorting.
15	Design and implement a console-based Snake and Ladders Game using Python. The game should simulate a turn-based board game played between 2 or more players on a 100-square board, incorporating snakes, ladders, dice roll mechanics, and a winner declaration. The objective is for a player to reach the 100th square before others, while avoiding snakes and utilizing ladders for quick progress.
16	Design and implement a Smart Text Editor using Python that supports not only basic text editing operations (like opening, editing, and saving files), but also intelligent features such as auto-correction, word/grammar suggestions, real-time word count, search and replace, and more.

Suggested List of Laboratory Experiments/Assignments	
17	Design and develop a Java-based application to automatically calculate the Term Work (TW) marks of students based on multiple academic and extracurricular parameters such as: Daily Attendance Unit Test / Prelim Performance Mock Practical Performance Student Achievements (e.g., participation in events, hackathons, etc.

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
PCC-255-CSD: Computer Graphics Laboratory		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical : 02 Hours/Week	01	**Oral :** 25 Marks **Termwork :** 25 Marks

Companion Course :Computer Graphics

Course Objectives: The course aims to:

To apply various methods and techniques for implementing line and circle drawing, transformation, animation and OpenGL using Object Oriented Programming.

Course Outcomes: Upon successful completion of this course, students will be able to:

CO1: Analyze and apply computer graphics algorithms for line-circle drawing, scan conversion and filling with the help of object oriented programming concepts.

CO2: Implement algorithms to clip line, to fill and clip polygons.

CO3: Apply logic to implement curves, fractals, animation and OpenGL concepts.

Course Contents

Guidelines for Instructor's Manual

The instructor's manual/Lab Manual is to be developed as a hands-on resource and reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of course, conduction and Assessment guidelines, topics under consideration-concept, objectives, outcomes, set of typical applications/assignments/guidelines, references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software and Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis. Program codes with sample output of all performed assignments are to be submitted as softcopy.

As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal may be avoided. Students programs maintained on cloud or college server by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory for accreditation purpose.

Guidelines for Laboratory/Term Work Assessment

Continuous assessment of laboratory work should be done based on overall performance and Laboratory assignments performance of student. Each Laboratory assignment assessment should be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include timely completion performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Laboratory Conduction

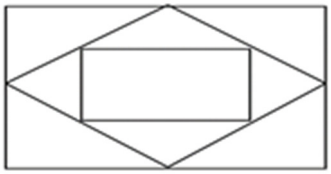
Use of open source software is encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned.

Operating System recommended: - 64-bit Open source Linux or its derivative

Programming tools recommended: - Open Source C++ / Java Programming tool like G++/JDK.

Set of suggested assignment list is provided in groups- A, B, C, D and E. Instructor is suggested to design lab assignments list by selecting/designing 8/9 suitable assignments (1/2 from each group) and a mini project.

Suggested List of Laboratory Experiments/Assignments

Sr.	Name of Assignment
	Group A
1	Write a C++ program to draw the following pattern using (a) the DDA line drawing algorithm for both rectangles and (b) Bresenham's line drawing algorithm for a diamond shape. 
2	Write a C++ program to modify the DDA line drawing algorithm to support different line styles (Dotted, Dashed, Solid, and Thick Lines).
3	Write a C++ program to draw a line using DDA and Bresenham's line drawing algorithms.
4	Write a C++ program using DDA and Bresenham's circle drawing algorithms to draw solid, dotted and dashed circles.
5	Write a C++ program using Bresenham's and Midpoint circle drawing algorithms to draw concentric circles.
	Group B
1	Write a C++ program to draw a concave polygon and fill it with the desired color using the scan fill algorithm.
2	Write a C++ program to draw a polygon and fill it with the desired color using flood fill and seed fill algorithms
3	Write a program to implement the Sutherland-Hodgeman algorithm for clipping any polygon. Provide the vertices of the polygon to be clipped and the pattern of clipping interactively.
4	Write a C++ program to implement the Cohen-Sutherland line clipping algorithm.
	Group C
1	Write a C++/Java program to implement translation, rotation and scaling of a 2D object about X axis and Y axis.
2	Write C++/Java program to implement translation, sheer, rotation and scaling transformations on equilateral triangle and rhombus.

3	Write a C++/Java program to implement rotation of a 2D object about X axis and an arbitrary point.
4	Write C++/Java program to draw implement Cube rotation about vertical axis passing through its centroid.
	Group D
1	Write a C++/Java program to generate fractal patterns using Koch curve.
2	Write a C++/Java program to generate fractal patterns using Snowflake.
3	Write a C++/Java program to generate a Hilbert curve using the concept of fractals.
4	Write a C++/Java program to generate a Bezier curve.
	Group E
1	Write a C++/Java program to create a moving car.
2	Write a C++/Java program to create a bouncing ball using a sinusoidal waveform.
3	Write a program to create a line in a window using OpenGL.
4	Write a C++/Java program to draw a 3D cube and perform following transformations on it using OpenGL: a) Scaling b) Translation c) Rotation about one axis.
5	Write OpenGL program to draw Sun Rise and Sunset.
6	Write a program using OpenMP to add two one-dimensional arrays.
	Group F
	Mini Project: (Suggested List: Not limited to -) • National Flag hoisting • Waves generated by water drop falling into the water • Kaleidoscope views generation (at least 3 colorful patterns) • Object such as flower, waves using any curve generation techniques

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
MDM-271-CSD: Computer Organization and Architecture		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 02Hours/Week	02	**CCE** : 30 Marks **End-Semester:** 70 Marks

Prerequisite Courses, if any :

1. Basics of Electronics Engineering
2. Digital Electronics

Companion Course if any: NA

Course Objectives: The course aims to:

1. To understand the structure, function and characteristics of computer systems.
2. To learn implementation of fixed-point operations and representation of floating-point numbers.
3. To understand processor organization and pipeline architecture.
4. To understand the hierarchical memory system and its organization
5. To study RISC and CISC architecture

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Understand the basics of Computer Organization and Architecture
- CO2: Identify and Apply arithmetic algorithms for solving ALU operations
- CO3: Illustrate processor organization and pipeline architecture
- CO4: Explore hierarchical memory system organization and apply mapping techniques for cache memory
- CO5: Compare CISC with RISC architecture and describe superscalar architecture.

Course Contents

Unit I -Introduction to Computer Organization and Architecture (06 Hours)

Computer Organization and Architecture, Structure and Function, Designing for Performance, , performance assessment. Computer Components, Computer Function, Interconnection structure, bus interconnection, Von Neumann Model, Harvard Architecture.

Unit II -Arithmetic Operations (06 Hours)

Arithmetic operations, Introduction to ALU: Definition, purpose, ALU components and importance of ALU in computer architecture.

Data Representation and Arithmetic Algorithms: Integer Data Computation- Addition and Subtraction, Multiplication: unsigned and signed multiplication, Booth's algorithm, Divisions: Restoring and Non-restoring algorithm.Floating point representation: IEEE 754 floating point number representation.

Unit III -Processor Organization (06 Hours)

Evolution of Intel processor architecture- 4 bit to 64 bit Processor Organization, Programmers model, Instruction cycle, Instruction format

Case Study-Intel 8086, Addressing modes and Formats- Addressing modes- immediate, direct, indirect, register, register indirect, displacement and stack, Instruction Pipelining- Pipelining strategy, pipeline performance, Performance measures: Speedup, Efficiency, Throughput, Pipeline Hazards.

Unit IV - Project Execution and Control (06 Hours)

Characteristics of Memory, Memory Hierarchy: cost and performance measurement.
Memory Hierarchy Properties: Inclusion, Coherency and Locality Cache Memory: Cache memory Concepts, Structure, Cache Operations, Design problems based on mapping techniques, Cache Coherency, Write Policies.Introduction to Associative memory and SCM (Storage Class Memory)

Unit V -RISC and CISC Processor Architecture (06 Hours)

Characteristics of RISC and CISC Processor, RISC vs CISC architecture, Use of Large Register File, Compiler - Based Register Optimization Superscalar Architecture, Features of Superscalar Architecture.

Learning Resources

Text Books:

1. W. Stallings, “Computer Organization and Architecture: Designing for performance||”, Pearson Education/ Prentice Hall of India, 2003, ISBN 978-93-325-1870-4, 7th Edition.
2. Zaky S, Hamacher, “Computer Organization”, 5th Edition, McGraw-Hill Publications, 2001, ISBN- 978-1-25-900537-5, 5th Edition.

Reference Books:

1. John P Hays, “Computer Architecture and Organization”, 3rd Edition,McGraw-Hill Publication, 1998, ISBN:978-1-25-902856-4.
2. Miles Murdocca and Vincent Heuring, “Computer Architecture and Organization- an integrated approach”, 2nd Edition, Wiley India Pvt. Ltd, ISBN:978-81-265-1198-3.
3. Tanenbaum, “Structured Computer Organization||”, 4th Edition,Prentice Hall of India, 1991 ISBN: 81 – 203 – 1553 – 7.
4. "Intel 8086/8088 User’s Manual" by Intel Corporation
5. Steve Furber, “ARM System On Chip architecture”, 2nd Edition, Pearson, ISBN-10: 8131708403.

MOOC / NPTEL/YouTube Links: -

1. Indranil Sengupta, Kamalika Datta , “ Computer Architecture and Organization”https://onlinecourses.n
   https://onlinecourses.nptel.ac.in/noc24_mg01/preview
2. V. Kamakot, “Computer Organization and Architecture”https://nptel.ac.in/courses/106106166
3. Smruti R.Sarangi, “ Computer Architecture”https://nptel.ac.in/courses/106102157

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
VSE- 281-CSD: Object Oriented Programming (JAVA)		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical: 04 Hours/Week	02	Term Work: 25 Marks Practical: 25 Marks

Prerequisite Courses : -Fundamentals of JAVA

Course Objectives: The course aims to:

To understand the basic concepts of Java Programming.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: To Develop Object-Oriented Programs.
- CO2: To Understand Java Fundamentals.
- CO3: To Develop Problem-Solving Skills.
- CO4: To Acquire practical knowledge of using common Java Standard Library and important packages.
- CO5: To Learn to handle runtime errors effectively using exception handling mechanisms in Java.

Course Contents

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be done based on overall performance and Laboratory assignments performance of student. Each Laboratory assignment assessment should be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Practical Examination

TBoth internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student's performance for advanced learning, understanding of the fundamentals, effective and efficient implementation. Consequently, encouraging efforts, transparent evaluation and fair approach of the evaluator will not create any uncertainty or doubt in the minds of the students. Therefore adhering to these principles will consummate our team efforts to the promising start of the student's academics.

Guidelines for Laboratory Conduction

TThe instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students. The instructor may set multiple sets of assignments and distribute among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged.

Operating System recommended: - 64-bit Open source Linux or its derivative

Programming tools recommended: - Java

Suggested List of Laboratory Experiments/Assignments	
	Group A (All are Compulsory)
1	Write a java program to perform the following operations: i) To check even or odd number. ii) To display the Fibonacci series.iii) To calculate Simple Interest.iv) Find maximum of three numbers.
2	Define a class to represent a bank account. Include the following members: Data members: - Name of depositor, Account number, Type of account, Balance amount in the account Member functions: - To assign initial values, to deposit an amount, to withdraw an amount after checking the balance, to display name & balance. Write a java program by using class and object.
3	Develop a java program that defines an abstract class called Shape, which includes two integer variables and an empty method named printArea(). Create three subclasses: Rectangle, Triangle, and Circle, each extending the Shape class. In each subclass, implement the printArea() method to calculate and display the area of the respective shape.
4	Create a Stud class to display student information by using a constructor and destructor. (Implement a default constructor, multiple constructors, a copy constructor, and an overloaded constructor).
	Group B (Any Four)
5	Write Programs in Java to sort i) an array in ascending order ii) to convert char Array to String.
6	Develop a Java program that declares an integer called day, where its value signifies a day of the week (ranging from 1 to 7) The program should then output the corresponding name of the day according to the value of day, using switch case statements.
7	Create a Java program that imports a user-defined package and utilizes the members of the classes contained in the package. Use Employee data.
8	Create a Java program that generates threads by extending the Thread class to print consecutive data.
9	Write a Java program to implement types of inheritance. Use library data.
10	Write a Java program to perform arithmetic operations of two complex numbers using class.
	Group C (Any Three)
11	Write a JAVA program to demonstrate exception handling mechanisms.
12	Write a Java program to read and display content of a file preceding line number.
13	Write a Java program to sort the given inputs by using function templates.
14	Write a Java program to store marks scored in subject “Object Oriented Programming” by N students in the class for the following operations: a.The average score of class b.Highest score and lowest score of class c.Count of students who were absent for the test d.Display mark with highest frequency
	Group D (Any One)
15	Design a mini project using Java which will use the different data structure with or without Java collection library and show the use of specific data structure on the efficiency (performance) of the code.
16	Design a mini project to implement a Smart text editor.

Learning Resources:-

Text Books:

- E Balagurusamy, “Programming with JAVA”, Tata McGraw Hill, 7th Edition.

• Herbert Schildt, “Java: The complete reference”, Tata McGraw Hill, 7th Edition.

Reference Books:

• Cay Horstmann , “Core Java Volume 1”, Kindle, 11th Edition.
• T. Budd, “Understanding OOP with Java”, Pearson Education, 2nd Updated Edition.

e-Books :

• “Introduction to Programming Using Java”, David J. Eck, Version 5.0, December 2006

MOOCs Courses:

• https://java-programming.mooc.fi/

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
AEC-282- CSD: Modern Indian Language (Marathi)		
Teaching /scheme	**Credits**	**Examination Scheme**
Tutorial : 01 Hour/Week **Practical** : 02 Hours/Week	01 01	**Term Work** : 25 Marks

Course Objectives: The course aims to:

अभ्यासक्रमाची उद्दिष्टे :

१. प्रगत भाषिक कौशल्यांची क्षमता विकसित करणे.
२. प्रसारमाध्यमांतील संज्ञापनातील स्वरूप आणि स्थान स्पष्ट करणे.
३. व्यक्तिमत्त्व विकास आणि भाषा यांच्यातील सहसंबंध स्पष्ट करणे.
४. लोकशाहीतील जीवनव्यवहार आणि प्रसारमाध्यमे यांचे परस्पर संबंध स्पष्ट करणे.
५. प्रसारमाध्यमांसाठी लेखनक्षमता विकसित करणे.

Course Contents

Unit I & II (07 Hours & 08 Hours)

घटक	तपशील
१	१. भाषा आणि व्यक्तिमत्त्व विकास : सहसंबंध २. लोकशाहीतील जीवनव्यवहार आणि प्रसारमाध्यमे
२	प्रसारमाध्यमांसाठी लेखन १ वृत्तपत्रासाठी बातमीलेखन आणि मुद्रितशोधन २ नभोवाणीसाठी भाषणाची संहितालेखन ३ दूरचित्रवाणीसाठी माहितीपटासाठी संहितालेखन

Case Study:

Unit III & IV (07 Hours & 08 Hours)

१	१. भाषा, जीवन व्यवहार आणि नवमाध्यमे, समाजमाध्यमे २. नवमाध्यमे आणि समाजमाध्यमांचे प्रकार : ब्लॉग, फेसबुक, ट्विटर. ३. नवमाध्यमे आणि समाजमाध्यमांविषयक साक्षरता, दक्षता, वापर आणि परिणाम
२	१. वेबसाईट आणि ब्लॉग, ट्विटरसाठी लेखन २. व्यावसायिक पत्रव्यवहार

Learning Resources

Text Books:

संदर्भ ग्रंथ :

१ सायबर संस्कृती, डॉ. रमेश वरखेडे
२ उपयोजित मराठी, संपादक डॉ. केतकी मोडक, संतोष शेणई, सुजाता शेणई
३ ओळख माहिती तंत्रज्ञानाची, टिमोथी जे. ओ लिअरी
४ संगणक, अच्युत गोडबोले, मौज प्रकाशन, मुंबई.
५ इंटरनेट, डॉ. प्रबोध चोबे, मनोरमा प्रकाशन, मुंबई.
६ व्यावहारिक मराठी, डॉ. ल. रा. नसिराबादकर, फडके प्रकाशन, कोल्हापूर.
७ आधुनिक माहिती तंत्रज्ञानाच्या विश्वात, शिक्रापूरकर दीपक, मराठे उज्ज्वल, उत्कर्ष प्रकाशन, पुणे.

Guidelines for Ability Enhancement Courses - Modern Indian Language (Marathi)

Implementation Guidelines

1. Subject teacher should frame minimum 08 assignments-based covering on all four units.
2. **Scaffolding:** Start with simpler tasks and gradually increase complexity. Provide necessary vocabulary and grammatical structures beforehand.
3. **Differentiation:** Offer varied levels of support for students with different proficiencies.
4. **Pair Work & Group Work:** Encourage collaborative learning and peer correction.
5. **Feedback:** Provide constructive feedback on all assignments, focusing on both accuracy and fluency.
6. **Authentic Materials:** Use real-world Marathi materials (simple songs, short videos, advertisements) as much as possible.
7. **Cultural Context:** Integrate cultural aspects into the assignments (e.g., describing a Marathi festival, a famous personality).
8. **Fun & Engaging:** Make the activities enjoyable to keep students motivated. Use games and competitive elements where appropriate.
9. **Technology Integration:** Use online dictionaries, translation tools (with caution), and Marathi typing tools.
10. These assignments can be adapted based on the students' proficiency level and the specific learning objectives of the Marathi course.

Suggested List of Assignments (Marathi):

1. **"Samvad Sadara Kara" (Present a Dialogue): Role-Playing Everyday Scenarios:** Objective is to practice conversational Marathi, understanding social cues. In pairs or small groups, students create and perform a short dialogue based on a given scenario.
2. **Vartamanpatra Vachan" (Newspaper Reading):** (Columns like Sports, political, finance, editorial, education, international news etc) in the daily Marathi newspapers, summarize and present in the practical. A summary should be added as part of the journal.
3. **Creative writing:** Write blogs and posts on social media up to 200 words on recent development in their field of study
4. **"Mala He Sangayche Aahe"** (I Want to Say This): Students are expected to show the objects and describe it to the class in Marathi. They should mention its color, size, use, why it's important to them, etc.
5. **Professional letter / report writing**
    a. Write a letter to the principal/director for organizing NSS camp in nearby village. Preparation of the budget, permission letters and report submission in the University
    b. Write a letter for internship sponsorship to any organization.
6. **Book Review** – Students are expected to read any novel, fiction or literature book of their choice and write a review on post it on social media of their choice.
7. **Participation in Competitions** (in college/out side the college) debate, declamation, elocution – A Report should be submitted
8. **Group Activity:** Road show, skit play, one-act play
9. **Participation in One-Act-Play -** Participation in Purushottam karandam, Firodia karandak, Dajikaka Gadgil Karandak and Shreetej Karandak.
10. **Marathi Film Review** – Watch the Marathi movie widely available on an OTT (Over-The-Top) platform, broadcaster in Television or availle on YouTube and write a review.

Savitribai Phule Pune University Second Year of Computer Science and Design (2024 Course)		
EEM-283-CSD: Engineering Product Design		
Teaching /scheme	**Credits**	**Examination Scheme**
Practical: 02 Hours/Week **Tutorial: 01 Hours/Week**	02	**Term Work :** 25 Marks

Prerequisite Courses, if any :

1. Basic knowledge of computer fundamentals and programming

Course Objectives: The course aims to introduce engineering students to the fundamentals of Digital electronics technology, enhance problem-solving abilities, and provide a strong foundation for careers in computing, automation, and embedded systems.

1. Apply and learn about the product design life cycle, giving particular attention to market demand and user needs.
2. Promote innovative thinking and ideation to address practical issues with a product-focused strategy.
3. Use both digital and physical tools to create low- to high-fidelity prototypes.
4. Incorporate multidisciplinary knowledge into product design, such as accessibility, ethics, cost-effectiveness, and sustainability.
5. Collaborate in groups to jointly create and showcase product concepts with functional models and supporting documentation.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1: Use the design thinking technique to identify and characterize user-centric problems and generate innovative product concepts.
- CO2: Using the proper tools, create and present working prototypes while taking accessibility, sustainability, and usability into account.
- CO3: Collaborate in groups to properly study, evaluate, and communicate the entire product design process in order to document and present it.

Course Contents

Unit I - Foundations of Product Design and Design Thinking (03 Hours)

Introduction to product design: Definitions, scope, and relevance in computer engineering, Evolution of product design in the digital age, Introduction to the Design Thinking Process: Empathize, Define, Ideate, Prototype, Test, Role of empathy in engineering design, Need-finding techniques: Observations, interviews, surveys, User personas and journey mapping.
Case study:
- Apple iPhone: User-centric product evolution
- Google Maps: Empathy in redesign for accessibility

Unit II - Ideation, Creativity and Innovation Strategies (03 Hours)

Techniques for ideation: Brainstorming, SCAMPER, Mind Mapping, Convergent vs Divergent thinking, Idea evaluation frameworks: Weighted Decision Matrix, NUF Test, Innovation principles: Frugal innovation, disruptive innovation, Enhancing creative confidence in engineering students.
Case study:

- Swiggy / Zomato UI Evolution: Ideation based on user feedback
- Tata Nano: Innovation with constraints

Unit III - Prototyping and Product Validation (03 Hours)

Importance of prototyping in the product development lifecycle, Types of prototypes: Low-fidelity vs High-fidelity, digital vs physical, Tools for prototyping: Canva, Figma, TinkerCAD, Arduino, Role of MVP (Minimum Viable Product) in product development, Basic user testing methods: Heuristic evaluation, A/B testing, surveys, Collecting, analyzing, and applying user feedback.

Case study:

- lipkart UX Revisions: Digital prototype testing
- Arduino-based Smart Irrigation: Low-cost IoT prototype

Unit IV - Sustainable, Inclusive and Ethical Design (03Hours)

Principles of inclusive and universal design, Designing for differently-abled and underserved communities,

Environmental impact and sustainability in design, Ethical design: Privacy, safety, and cultural sensitivity, Regulatory and legal aspects (e.g., BIS, ISO, accessibility standards).

Case Study:

- Microsoft Adaptive Accessories – Assistive Technology for All
- Jaipur Foot: Inclusive, affordable prosthetic design

Unit V -Documentation, Communication and Reflection in Product Design (03 Hours)

Purpose and components of product design documentation, Methods of visually representing designs: Sketches, infographics, storyboards, Creating effective pitch decks and presentations, Writing technical reports and reflective journals, Role of peer review, feedback, and iteration.

Case study:

- IDEO Project Methodology: Design documentation walkthrough
- SmartBin : Effective product pitch for smart waste bins

Learning Resources

- **Text Books:**

1. Karl T. Ulrich, Steven D. Eppinger, Maria C. Yang, “Product Design and Development”, McGraw-Hill Education, ISBN: 9781259708581
2. Idris Mootee, “Design Thinking for Strategic Innovation”, Wiley, ISBN: 9781118620120
3. Clive L. Dym, Patrick Little, Elizabeth Orwin, “Engineering Design: A Project-Based Introduction”, Wiley, ISBN: 9781118324585

- **Reference Books:**

1. Tom Kelley (IDEO), “The Art of Innovation”, Crown Business, ISBN: 9780385499842
2. Tim Brown (IDEO), “Change by Design”, Harper Business, ISBN: 9780061766084
3. David Kelley, Tom Kelley, “Creative Confidence: Unleashing the Creative Potential Within Us All”, Crown Business, ISBN: 9780385349369
4. Don Norman, “Design of Everyday Things”, Basic Books, ISBN: 9780465050659

- **MOOC / NPTEL/YouTube Links: -**

1. Prof. J. Ramkumar, Prof. Amandeep Singh, IIT Kanpur, “Product Design and Manufacturing”[NPTEL Link](https://onlinecourses.nptel.ac.in/noc21_me66/preview)

2. Prof. Ashwin Mahalingam, Prof. Bala Ramadurai, IIT Madras, "Design Thinking - A Primer"[NPTEL Link](https://onlinecourses.nptel.ac.in/noc22_mg32/preview),

3. Prof. Rajiv Ratn Shah, IIIT Delhi, "Human-Computer Interaction"[NPTEL Link](https://onlinecourses.n

4. Prof. Supradip Das, Prof. Swati Pal, Prof. Debayan Dhar, IIT Guwahati ,"Product Design Innovation," [NPTEL Link](https://onlinecourses.nptel.ac.in/noc21_de01/preview)

Course Contents

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a hands-on resource and reference. The instructor's manual needs to include a prologue (about the university/program/institute/department/foreword/preface), the university syllabus, conduction & Assessment guidelines, topics under consideration, concept objectives, outcomes, a set of typical applications/assignments/guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by the student in the form of a journal. The journal consists of a prologue, certificate, table of contents, and write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software & Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept, instructions/features used, test cases, conclusion/analysis, and references). Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listings to journals may be avoided. Use of DVD containing student programs maintained by Laboratory Incharge is highly encouraged. For reference, one or two journals may be maintained with program prints at Laboratory

Guidelines for Laboratory/Term Work Assessment

Continuous assessment of laboratory work is based on overall performance and Laboratory assignments performance of the student. Each Laboratory assignment assessment will assign a grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include timely completion, performance, innovation, efficient codes, punctuality, and neatness.

Suggested List of Assignment (Any SIX)	
1	Ideation and Problem Identification: Identify a real-world problem in the domains of healthcare, agriculture, education, or urban living that can be addressed through a technological product. Tools: Mira, Figma, or Canva
2	User-Centred Design and Wireframing: Develop wireframes for a mobile or web application focusing on user experience and interface design. Tools: Figma, Adobe XD
3	Rapid Prototyping with IoT Integration: Create a functional prototype of a smart device (e.g., smart irrigation system, health monitoring wearable) integrating sensors and microcontrollers. Tools: Arduino, Raspberry Pi, Tinkercad
4	Design for Sustainability: Redesign an existing electronic product to enhance its sustainability by focusing on energy efficiency, recyclability, and minimal environmental impact. Tools: AutoCAD, SolidWorks, or Fusion 360
5	Human-Computer Interaction (HCI) Evaluation: Conduct usability testing on a software application to assess its user-friendliness and accessibility. Tools: UsabilityHub, Google Forms, or Hotjar
6	Value Engineering and Cost Analysis: Analyze the cost components of a tech product and propose design modifications to reduce costs without compromising quality. Tools: Excel, Costimator, or custom spreadsheets
7	Inclusive Design Challenge: Design a product interface that is accessible to users with disabilities, ensuring compliance with accessibility standards. Tools: WAVE, Axe, or Lighthouse
8	Ethical and Legal Aspects in Product Design: Understand and evaluate the ethical, legal, and societal implications of a tech-based product. Tools: Word processors, Canva for presentations

Learning Resources

Text Books:

1. Dr. M. A. Bulsara, Dr. H. R. Thakkar, Charotar Publishing House Pvt. Ltd., 2nd Edition 2015 (Revised & Enlarged) ISBN : 9789385039140
2. Product Design for Engineers by Devdas Shetty, Cengage Publishing, ISBN: 9788131533031

Reference Books:

1. Product Design and Manufacturing – A.K. Chitale & R.C. Gupta
2. Product Design for Engineers – Devdas Shetty
3. The Design of Everyday Things—Don Norman
4. IDEO Design Kit – www.designkit.org
5. Ethical OS Toolkit – www.ethicalos.org
6. CS3216 Product Design Projects – www.cs3216.com

Online Resources:

1. [IDEO.org Design Kit](https://www.designkit.org/), Design Thinking tutorials, Empathy, Ideation, Prototyping
2. [Interaction Design Foundation](https://www.interaction-design.org/), HCI & UX Design Learning, Assignments on usability and evaluation
3. [TinkerCAD](https://www.tinkercad.com/), Online prototyping and circuit simulation, Prototyping with Arduino, IoT,

4. [Figma](https://www.figma.com/), Wireframing & UI Design, Assignments on user-centered design
5. [MIT D-Lab](https://d-lab.mit.edu/), Sustainable design & inclusive innovation, Assignments on design for sustainability and inclusion
6. [Canva](https://www.canva.com/), Design mockups and visuals, Sketches and presentation of product ideas

Savitribai Phule Pune University Second Year of Computer Science and Design(2024)		
VEC-284- CSD - Environmental Studies		
Teaching /scheme	**Credits**	**Examination Scheme**
Theory : 02 Hours/Week	02	**CCE :** 15 Marks **End-Sem Examination :** 35 Marks

Course Objectives: The course aims to:

1. To introduce the multidisciplinary nature and scope of environmental studies.
2. To understand ecosystem structures, biodiversity, and ecological balance through hands-on observation and documentation.
3. To examine the use and impact of natural resources on environmental sustainability.
4. To explore biodiversity conservation practices and develop eco-sensitive thinking through field-based inquiry.

Course Outcomes: Upon successful completion of this course, students will be able to:

- CO1. **Illustrate** the interdependence of ecosystems through activity-based exploration
- CO2. **Analyze** the role of natural resources in sustainable development using real-world data.
- CO3. **Investigate** biodiversity threats and conservation strategies through surveys and projects
- CO4. **Create** awareness tools or **reports** promoting sustainability based on their findings.

Course Contents

Unit I - Environment and its issues (07 Hours)

a) Environment Meaning of Environment, Types of Environment, Components of Environment,
b) Man- Environment relationship, importance of environment,
c) Need for Public Awareness
d) Ecosystem-Meaning, Major Components of Ecosystem
e) Case studies of Forest Ecosystem, Grassland Ecosystem, Desert Ecosystem, Aquatic Ecosystem
f) Stability of Ecosystem in Sustainable Environment

Unit III - Environment Pollution (07 Hours)

a) Definition of Pollution, Types of Pollution
b) Air Pollution-Meaning, Sources, effects of air pollution, Air Pollution Act
c) Water Pollution Meaning, Sources, Effects of Water pollution, Water Pollution Act
d) Noise Pollution Meaning, Sources, Effect of Noise Pollution
e) Solid Waste Pollution Meaning, sources, Effect of Waste Pollution

Unit III - E-Waste Managements and Acts (08 Hours)

E- waste; composition and generation. Global context in e- waste; E-waste pollutants, E waste hazardous properties, Effects of pollutant (E- waste) on human health and surrounding environment, domestic e-waste disposal, Basic principles of E waste management, Technologies for recovery of resources from electronic waste, resource recovery potential of e-waste, steps in recycling and recovery of materials-mechanical processing, technologies for recovery of materials, occupational and environmental health perspectives of recycling e-waste in India.

Unit IV - E-waste Control and measures (08 Hours)

Need for stringent health safeguards and environmental protection laws in India, Extended Producers Responsibility (EPR), Import of e-waste permissions, Producer-Public-Government cooperation,

Administrative Controls & Engineering controls, monitoring of compliance of Rules, Effective regulatory mechanism strengthened by manpower and technical expertise, Reduction of waste at source

Practical Assignments

Week	Topic to be covered
1	Introduction : Group discussion and poster making on "Why Environmental Studies Matter for Technologists"
2	Eco Mapping: Identify and document elements of an ecosystem within the college campus
3	Model the Food Web: Create food chains and food webs using flowcharts (digital tools like Canva / Lucid chart)
4	Case Study Review: Present real-world examples of forest, grassland, and aquatic ecosystems
5	Soil and Water Testing Activity: Test soil pH, water quality (use school-level kits), and interpret results
6	Field Visit / Virtual Tour: Document deforestation or mining impact in a chosen region; students prepare a comparative report
7	Water Audit Exercise: Estimate water usage at home/hostel and identify areas of overuse; propose conservation measures
8	Renewable Energy Models: Create a simple model or PPT on any renewable energy source (e.g., solar cooker, wind energy demo)
9	Biodiversity Documentation: Survey nearby areas for plant/animal species; identify any endemic/endangered species
10	Conservation Proposal Pitch: In groups, students prepare a mini proposal for biodiversity conservation at local level
11	Group Project Work: Work on mini project report/documentation on any ecosystem/natural resource/e-waste management topics
12	Presentation & Viva: Final presentation and oral examination based on project work and learning portfolio

Learning Resources

Text Books:

1. Odum, Eugene P. “Fundamentals of Ecology”
2. R. Rajagopalan, “Environmental Studies – From Crisis to Cure”, Oxford
3. Johri R., E-waste: implications, regulations, and management in India and current global best practices, TERI Press, New Delhi

Reference Books:

1. Erach Bharucha, “Textbook of Environmental Studies”, UGC
2. Anubha Kaushik and C.P. Kaushik, “Environmental Studies”, New Age International

E-Books Links: -

1. https://www.environment.gov.in
2. https://www.unep.org
3. https://news.mit.edu/2013/ewaste-mit

Savitribai Phule Pune University, Pune

Maharashtra, India

Task Force for Curriculum Design and Development

Programme Coordinator

Dr. D.B. Kshirsagar - Member, Board of Studies - Computer Engineering

Team Members for Course Design

Dr. Prashant Yawalkar	Dr. Ravindra. S. Tambe
Dr. Prakash Kalavadekar	Prof. Narayan B. Vikhe
Dr. Jayashri S.Hase	Prof. Vidya V. Jagtap
Dr. Balaji Bodkhe	Dr. Vandana Navale
Prof. Shubhangi Said	

Chairman

Dr. Nilesh Uke - Board of Studies Computer Engineering

Savitribai Phule Pune University, Pune

Dean

Dr. Pramod Patil - Dean – Science and Technology
Savitribai Phule Pune University, Pune

***