Marathwada Mitra Mandal's
COLLEGE OF ENGINEERING, Karvenagar, Pune - 52
An Autonomous Institute affiliated to SPPU
Website: www.mmcoe.edu.in email id: mmcoe@mmcoe.edu.in

Curriculum Structure Sem-III & IV

Second Year B.Tech Computer Engineering Semester-III

Course Code	Course Name	Course Type	Teaching Scheme (Hrs/week)			Examination Scheme						Credits			
			L	P	T	CIE	ETE	TW	PR	OR	Total	L	P	T	Total
CE24PCC201	Data Structures	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC202	Object Oriented Programming	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC203	Digital Electronics and Computer Organization	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24MDM204	*Multidisciplinary Minor Theory	MDM	2	-	-	40	60	-	-	-	100	2	-	-	2
CE24PCC205	Data Structures Laboratory	PCC	-	2	-	-	-	-	50	-	50	-	1	-	1
CE24PCC206	Object Oriented Programming Laboratory	PCC	-	2	-	-	-	-	50	-	50	-	1	-	1
CE24MDM207	*Multidisciplinary Minor Laboratory	MDM	-	2	-	-	-	25	-	-	25	-	1	-	1
CE24CEP208	Project based Learning	CEP	-	4	-	-	-	50	-	-	50	-	2	-	2
CE24EEM209	Human Values and Professional Ethics	EEM	-	-	2	-	-	50	-	-	50	-	-	2	2
SH24VEC201	Environment Sustainability	VEC	-	-	2	-	-	50	-	-	50	-	-	2	2
Total			11	10	4	160	240	175	100	-	675	11	5	4	20

* Refer separate MDM Booklet

Second Year B.Tech Computer Engineering Semester-IV

Course Code	Course Name	Course Type	Teaching Scheme (Hrs/week)			Examination Scheme						Credits			
			L	P	T	CIE	ETE	TW	PR	OR	Total	L	P	T	Total
CE24PCC251	Database Management Systems	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC252	Principles of Programming Languages	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC253	Software Engineering	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24OEL254	* Open Elective	OEL	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC255	Database Management Systems Laboratory	PCC	-	2	-	-	-	-	50	-	50	-	1	-	1
CE24PCC256	Principles of Programming Languages Laboratory	PCC		2					50	-	50	-	1	-	1
CE24VSE257	Web Technology	VSE	-	4	-	-	-	50	-	-	50	-	2	-	2
CE24VEC258	OS Workshop	VEC	-	-	2	-	-	50	-	-	50	-	-	2	2
SH24AEC202	Professional Communication and Personality Development	AEC	-	-	2	-	-	50	-	-	50	-	-	2	2
Total			12	8	4	160	240	150	100	-	650	12	4	4	20

L- Lecture
ETE- End Term Examination
L : 1 Hr.= 1 credit

P- Practical
TW- Term work
P: 2 Hr. = 1 Credit

T- Tutorial
PR- Practical
T: 1 Hr. = 1 Credit

CIE-Continuous Internal Evaluation
OR- Oral

BOS Chairperson
(Dr. K. S. Thakre)

Dean Academics
(Dr. B. P. Vasgi)

Principal
(Dr. S. R. Sakhare)

'येथे बहुतांचे हित'

Marathwada Mitra Mandal's

COLLEGE OF ENGINEERING

Karvenagar, Pune - 52.

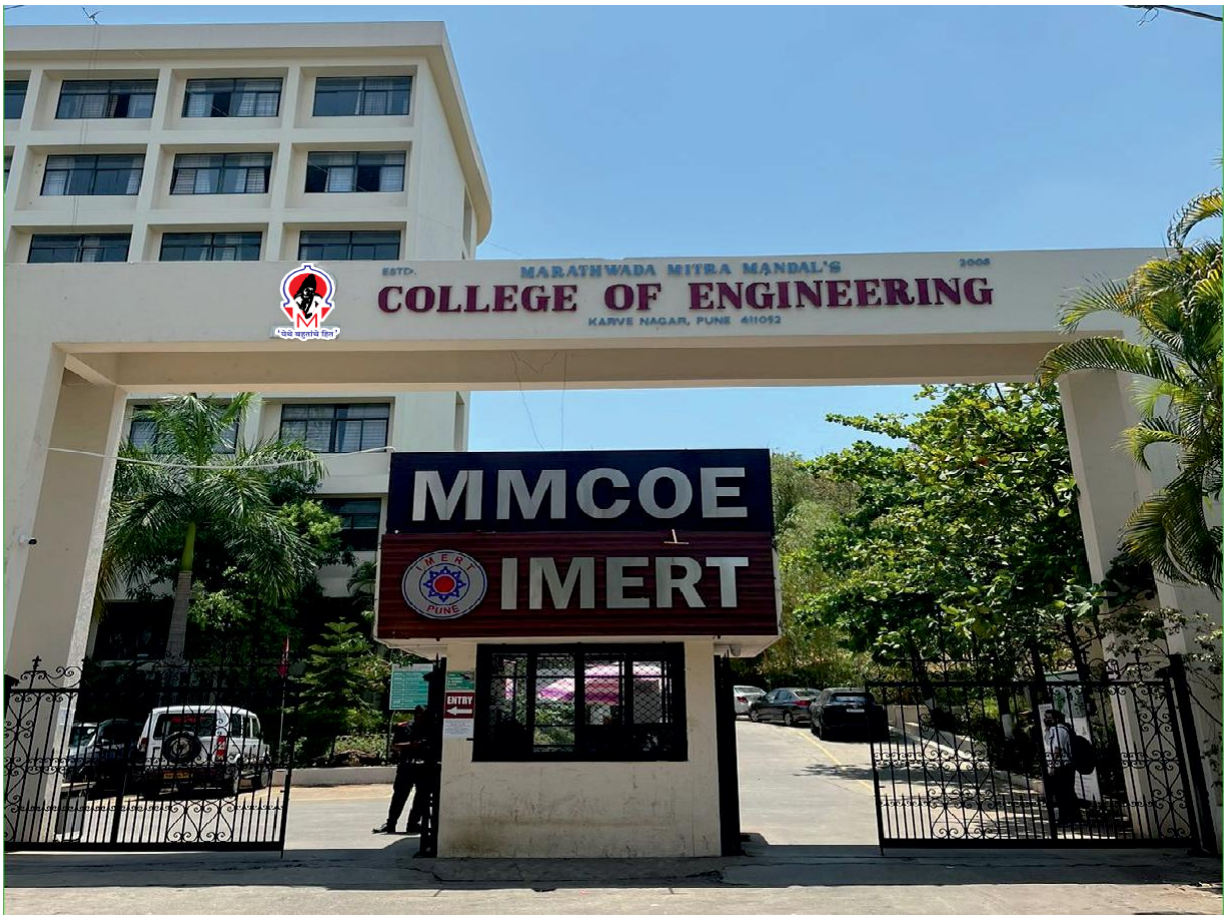
An Autonomous Institute affiliated to SPPU

Accredited with A++ Grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956

www.mmcoe.edu.in

Curriculum Structure and Syllabus

Second Year B.Tech. Computer Engineering



Academic Year 2025-26

Marathwada Mitra Mandal's College of Engineering

Karvenagar, Pune- 52

An Autonomous Institute

"येथे बहुतांचे हित"

Curriculum for

Second Year B. Tech Computer Engineering

A.Y. 2025-26

Permanently Affiliated to SPPU | Accredited with A++ grade by NAAC
Recipient of Best College award by SPPU | Accredited by NBA
Recognized under 2(f) and 12(B) of UGC Act 1956
www.mmcoe.edu.in

CONTENTS

Institute Vision

To be a globally renowned institution through excellence in engineering education for sustainable and holistic development

Institute Mission

M1: Empower students with cutting-edge technologies and global competencies

M2: Foster culture of research and entrepreneurial mindset

M3: Imbibe social and professional values

M4: Provide an inclusive environment for lifelong learning

Department Vision

To excel in computer engineering education for sustainable development.

Department Mission

M1: To develop globally competent professionals through interdisciplinary learning and Center of Excellence**.**

M2: To empower research, innovation, and entrepreneurial thinking

M3: To promote ethical values and holistic development

M4: To prepare graduates for lifelong learning and dynamic careers

Knowledge and Attitude Profile (WK)

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.

Program Outcomes (PO)

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

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

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

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

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

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

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

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

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

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

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Program Educational Objectives (PEO)

PEO1: Graduates will demonstrate strong foundational knowledge to address global and multidisciplinary challenges

PEO2: Graduates will engage in innovation, research, and entrepreneurship across multidisciplinary domains

PEO3: Graduates will practice ethical conduct, professional and social responsibility

PEO4: Graduates will exhibit lifelong learning and continuous skill enhancement

Program Specific Outcomes (PSOs)

A Graduate of the Computer Engineering Program will be able to

PSO1: Apply problem solving skills to design effective solutions in High Performance Computing, Artificial Intelligence and Cyber Security

PSO2: Develop advanced skill-based solutions using standard Software Engineering practices

Abbreviations

BSC:	Basic Science Course
UG:	Undergraduate Programme
ESC:	Engineering Science Course
PCC:	Program Core Courses
PEC:	Program Elective Courses
MDM:	Multidisciplinary Minor Courses
OEL:	Open Elective
VSE:	Vocational & Skill Enhancement Course
AEC:	Ability Enhancement Course
EEM:	Entrepreneurship/Economics/Management
IKS:	Indian Knowledge System
VEC:	Value Education Course
RMD:	Research Methodology
CEP/FPR:	Comm. Eng. Project (CEP)/Field Project (FP)
PRJ:	Project
INT/OJT:	Internship/On-Job Training
CCC:	Co-Curricular Courses
IT:	Internal Tool
ET:	External Tool
ETE:	End –Term Examination
CIE:	Continuous Internal Evaluation
TW:	Term work
OR:	Oral Examination
PR:	Practical Examination
L:	Lecture
P:	Practical
T:	Tutorial
OL:	Online Teaching
ODL:	Open Distance Learning

Assessment Tool Guidelines for Continuous Internal Evaluation (CIE)

AY: 2025-26

Distribution of CIE 40 marks for each subject is as below:

Sr. No.	Unit No.	Weightage	Assessment Tools
1	Unit 1	8 Marks	Class Test
2	Unit 2	8 Marks	
3	Unit 3	8 Marks	1. Quizzes 2. Home Assignments 3. Case Study 4. Field work 5. Report writing 6. Presentations/Seminar Topics 7. Mini projects/Course Projects 8. Mind map 9. Poster Presentation 10. Problem Solving and Coding 11. Parametric Study 12. Any other assessment tool with permission of BoS chairperson
4	Unit 4	8 Marks	
5	Unit 5	8 Marks	

1. Above are the suggestive assessment tools for CIE.
2. The class test is a one-time assessment; no retests will be conducted
3. Module coordinators should review and verify the assessment tools implemented for each course.
4. Course coordinators are required to maintain softcopy records of the respective CIE assessment tools used for their classes.

Curriculum Structure Sem-III & IV

Second Year B.Tech Computer Engineering Semester-III															
Course Code	Course Name	Course Type	Teaching Scheme (Hrs/week)			Examination Scheme						Credits			
			L	P	T	CIE	ETE	TW	PR	OR	Total	L	P	T	Total
CE24PCC201	Data Structures	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC202	Object Oriented Programming	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC203	Digital Electronics and Computer Organization	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24MDM204	*Multidisciplinary Minor Theory	MDM	2	-	-	40	60	-	-	-	100	2	-	-	2
CE24PCC205	Data Structures Laboratory	PCC	-	2	-	-	-	-	50	-	50	-	1	-	1
CE24PCC206	Object Oriented Programming Laboratory	PCC	-	2	-	-	-	-	50	-	50	-	1	-	1
CE24MDM207	*Multidisciplinary Minor Laboratory	MDM	-	2	-	-	-	25	-	-	25	-	1	-	1
CE24CEP208	Project based Learning	CEP	-	4	-	-	-	50	-	-	50	-	2	-	2
CE24EEM209	Human Values and Professional Ethics	EEM	-	-	2	-	-	50	-	-	50	-	-	2	2
SH24VEC201	Environment Sustainability	VEC	-	-	2	-	-	50	-	-	50	-	-	2	2
Total			11	10	4	160	240	175	100	-	675	11	5	4	20

* Refer separate MDM Booklet

Second Year B.Tech Computer Engineering Semester-IV															
Course Code	Course Name	Course Type	Teaching Scheme (Hrs/week)			Examination Scheme						Credits			
			L	P	T	CIE	ETE	TW	PR	OR	Total	L	P	T	Total
CE24PCC251	Database Management Systems	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC252	Principles of Programming Languages	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC253	Software Engineering	PCC	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24OEL254	* Open Elective	OEL	3	-	-	40	60	-	-	-	100	3	-	-	3
CE24PCC255	Database Management Systems Laboratory	PCC	-	2	-	-	-	-	50	-	50	-	1	-	1
CE24PCC256	Principles of Programming Languages Laboratory	PCC		2					50	-	50	-	1	-	1
CE24VSE257	Web Technology	VSE	-	4	-	-	-	50	-	-	50	-	2	-	2
CE24VEC258	OS Workshop	VEC	-	-	2	-	-	50	-	-	50	-	-	2	2
SH24AEC202	Professional Communication and Personality Development	AEC	-	-	2	-	-	50	-	-	50	-	-	2	2
Total			12	8	4	160	240	150	100	-	650	12	4	4	20

L- Lecture

P- Practical

T- Tutorial

CIE-Continuous Internal Evaluation

ETE- End Term Examination

TW- Term work

PR- Practical

OR- Oral

L : 1 Hr.= 1 credit

P: 2 Hr. = 1 Credit

T: 1 Hr. = 1 Credit

SEMESTER III

Second Year B.Tech Computer Engineering

Semester-III

Course Code: CE24PCC201 **Course Name: Data Structures**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	3

Prerequisite: Fundamentals of Programming Languages (CE24PCC101)

Course Objectives:

The course aims to:

- Demonstrate the use of arrays for storing and accessing elements using index-based operations.
- Explain the need for sorting techniques in optimizing data retrieval.
- Construct and manipulate singly and doubly linked lists to perform insertion and deletion operations.
- Design tree-based structures such as binary search trees for hierarchical data representation.
- Apply tree and graph algorithms to find optimal paths in a network.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Implement basic operations on linear data structure
CO2: Implement searching, sorting and hashing techniques for solving real life problems
CO3: Develop programs using linked lists for solving problems
CO4: Apply concepts of stack and queue data structures to real life problems
CO5: Develop programs using nonlinear data structures for solving computing problems

Unit	Contents	Duration (Hrs.)
1	**Introduction to Data Structures:** Data, Information, Knowledge, and Data structure, Abstract Data Types (ADT), Data Structure Classification, Characteristics of algorithm, Design tools of algorithm, analysis of algorithm using step count method and growth rate. **Array:** Sequential Organization, Array as an ADT, Operations, Storage Representation, Multidimensional Arrays, Sparse matrix representation, Simple and Fast Transpose of Sparse matrix, Time and Space tradeoff, Applications.	7
2	**Searching , Sorting and Hashing:** **Searching:** Linear and Binary Search **Sorting methods**: Bubble sort, Insertion Sort, Selection sort, Radix Sort. **Hashing:** Hash table, Hash functions, Properties of good hash function, Collision resolution strategies- open addressing and chaining, Applications.	7
3	**Linked List:** Static and Dynamic Allocation, Representation using dynamic memory management, Types: singly, Circular, Doubly, Primitive Operations - Create, Traverse, Search, Insert, Delete, Applications.	8
4	**Stack and Queue:** **Stack:** Concept, Representation using Sequential Organization, Operations (PUSH and POP), Applications: Polish notations, Expression Conversion and Evaluation, Recursion- concept, variants of recursion- direct, indirect, tail and tree.	8

	Queue: Concept, Representation using Sequential organization, Operations, Circular Queue,Double ended queue,Types of Double ended queue , Priority Queue, Applications.	
5	**Tree and Graph:** **Tree:** Tree Terminology, representation using array and linked list, Types, Tree traversal, Binary Search tree (BST): Insertion, deletion, Height Balanced Tree-AVL tree , B Tree, Applications. **Graph:** Graph Terminology, Storage representation, Adjacency matrix, adjacency list, Traversals-depth first and breadth first search, Minimum spanning tree: Prim's and Kruskal's algorithm,Applications.	10
	Total Hours	**40**

Text Books

1. Sartaj Sahni, Ellis Horowitz, "Fundamentals of Data Structures in C", Orient blackswan, 2nd Edition, 2010, ISBN 81- 7515-257-5.
2. Michael T. Goodrich, Roberto Tamassia, David M. Mount, "Data Structures and Algorithms in C++", ISBN-13 978-8126512607 Publisher WILEY INDIA EDITION, Publication date 1 January 2015.

Reference Books

1. E. Balagurusamy, "Object Oriented Programming with C++", Eighth Edition, ISBN-13 978-9389949186, McGraw Hill, 24 September 2020
2. Yashavant Kanetkar, "Let us C++", BPB Publications (16 September 2020), ISBN-10 : 9388176642, ISBN-13 : 978-9388176644.
3. G.A.V. PAI, "Data Structures and Algorithms, Concepts, Techniques and Applications", Tata McGraw-Hill, Volume1 1st Edition, 2017. ISBN-10: 0070667268/ ISBN-13: 978-0070667266.
4. Richard F. Gilberg& Behrouz A. Forouzan, "Data Structures, Pseudo code Approach with C", Cengage Learning India Edition, 2nd Edition, 2007, ISBN 10: 8131503143 ISBN 13: 9788131503140.
5. Y. Langsam, M. Augenstinand, A. Tannenbaum,"Data Structures using C & C++",Pearson Education India, Second Edition,2015, ISBN 10: 9332549311, ISBN 13: 978-9332549319.

Online References

1. NPTEL Course: Prof. Naveen Garg, IIT Delhi, "Data Structures And Algorithms" (Link: https://nptel.ac.in/courses/106102064)
2. https://www.cs.usfca.edu/~galles/visualization/Algorithms.html
3. https://visualgo.net/en

Second Year B.Tech Computer Engineering

Semester- III

Course Code: CE24PCC202 **Course Name: Object Oriented Programming**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3		-	-	-	40	60	-	-	-	100	3		-	3

Prerequisite: Fundamentals of Programming Languages (CE24PCC101)

Course Objectives:

The course aims to:

- Understand the concepts of data abstraction, encapsulation, and object oriented modeling.
- Apply code reusability principles using inheritance.
- Understand how virtual functions enable dynamic binding to achieve polymorphism.
- Implement the concepts of exception and file handling.
- Design generic classes with C++ templates and Standard Template Library.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand key concepts of object-oriented programming.
CO2: Apply reusability principle using inheritance for object oriented design.
CO3: Design solutions for real life problems using polymorphism concepts.
CO4: Apply exception and file handling concepts in real world applications.
CO5: Implement generic programming using template and STL in OOP applications.

Unit	Contents	Duration (Hrs.)
1	**Introduction to OOP and Basic C++ Concepts** **Introduction to OOP:** Overview of programming languages, Need and evolution of OOP, Key features of OOP, Applications of OOP. **Classes and Objects:** Declaring and defining classes, Specifying Member functions and data members, object creation, Memory allocation for objects, Access specifiers, Static members, friend functions. **Function Prototype, Constructors and Destructors:** Constructor types, Overloading of constructors, Destructor	**8**
2	**Inheritance and Pointers** **Inheritance:** Need, Base and Derived classes, Types of inheritance, Access control in inheritance, Constructor and destructor in Derived Class, diamond problem (Virtual Base Class), Abstract class, Friend Class, Nested Class. **Pointers:** Declaration and initialization of pointers, indirection Operators, new & delete operator, Pointers to objects, this pointer, Arrays of Pointers, Function pointers, Pointers to Pointers, Pointers to Derived classes, Null pointer, void pointer.	8
3	**Polymorphism:** Introduction, Types, Operator Overloading, Overloading Unary and Binary Operators, Function overloading, Data Conversion, Type casting, Keywords explicit and mutable. **Run Time Polymorphism-** Pointers to Base class, virtual function, pure virtual function and virtual table, virtual destructor, abstract base class.	8
4	**Files:** Classes for file stream operation, Opening and closing a file, File modes, Error Handling functions in file, File Pointers and Their Manipulation, File	8

	Operations on Characters, File Operations on Binary Files, Sequential input & output operations, Updating Random access file **Exception Handling:** Exception Handling mechanism using try, throw & catch, Divide by Zero exception, Multiple catch statements, re-throwing an exception, specifying exceptions, user defined exceptions, constructor, destructor and exception handling, exception and inheritance.	
5	**Templates and Standard Template Library (STL)** **Templates:** Introduction to Generic Programming, Function Template and Class templates, Template with multiple parameters, Overloading of template functions, Non-type template argument **Standard Template Library (STL):** Introduction to STL Components, Containers, Algorithms and Iterators, Application of Container classes: vector, list, map.	8
	Total Hours	**40**

Text Books

1. E. Balagurusamy, "Object -Oriented Programming with C++", McGraw Hill Education, Eighth Edition, Sept. 2020, ISBN-13 : 978-9389949186.
2. Robert Lafore, "OOP in C++", Pearson Publishing, 4th Edition, 2001, ISBN:0672323087 (ISBN 13: 9780672323089).
3. Yashwant Kanetkar, "Let Us C++", BPB Publications, ISBN: 9789388176644

Reference Books

1. Bjarne Stroustrup, The C++ Programming language, Third edition, 2008, Pearson Education. ISBN 9780201889543.
2. Deitel, C++ How to Program, 4th Edition, Pearson Education,2002, ISBN:81-297-0276-2.
3. Herbert Schildt, C++ The complete reference, Eighth Edition, McGraw Hill Professional, 2011, ISBN:978-00- 72226805

Online References

1. NPTEL Course: Prof. Partha Pratim Das, IIT Kharagpur, "Programming in Modern C++" (Link: https://onlinecourses.nptel.ac.in/noc24_cs44/preview)
2. NPTEL Course: By Prof. Partha Pratim Das, IIT Kharagpur. "Programming in C++" (Link: https://onlinecourses.nptel.ac.in/noc21_cs02/preview)

Second Year B.Tech Computer Engineering

Semester- III

Course Code: CE24PCC203 Course Name: Digital Electronics and Computer Organization

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3		-	-	-	40	60	-	-	-	100	3	-	-	3

Prerequisite: Basics of Electronics Technology (ET24PCC101), Basics of Electronics Technology Tutorial (ET24PCC102)

Course Objectives:

The course aims to:

- Design simplified Boolean expressions using Karnaugh Maps to enhance logical circuit efficiency.
- Construct combinational logic circuits such as multiplexers, decoders, and adders using Boolean logic.
- Apply the concepts of sequential digital circuits to design systems using flip-flops, counters, and shift registers.
- Implement assembly language programs by applying instruction sets and addressing modes.
- Apply the concepts of memory hierarchy and I/O organization to understand and analyze computer system architecture.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Design K Map for minimization of Boolean Expressions.

CO2: Apply the knowledge of combinational logic design.

CO3: Apply the knowledge of sequential digital circuits.

CO4: Apply addressing modes and instruction set to implement assembly language programs.

CO5: Use the concept of I/O and memory organization for understanding computer systems.

Unit	Contents	Duration (Hrs.)
1	**Combinational Circuits** Minimization of Boolean function using K-map, Minimization of Product of Sum(POS) and Sum of Product(SOP) functions, don't care conditions,Introduction to Numbers and Codes, BCD, Gray, Excess-3 and their applications, Code conversion using logic gates, Integer and floating point number representation, Signed and unsigned numbers, arithmetic operations.	8
2	**Combinational Logic Design** Realization of combinational functions like comparison, decoding, multiplexing, demultiplexing, Design of Half Adder and Full Adder, Design of Half Subtractor and Full Subtractor, BCD Adder, Look ahead and carry generator, Introduction to Carry Propagation Adder.	8
3	**Sequential Logic Design** Flip-Flop: SR, JK,D,T, Preset and Clear, Master Slave JK Flip Flops, Truth Tables and Excitation tables, Conversion from one type to another type of Flop-Flop. Registers: SISO, SIPO, PISO, PIPO, Shift Registers, Bidirectional Shift Register, Ring Counter , Universal Shift Register Counters: Asynchronous Counter, Synchronous Counter, Modulus of the counter (IC 7490), Sequence Generator.	8
4	**Introduction to Computer Organization** Function and structure of a computer, Functional components and Interconnection, Register organization, General purpose registers, Organization and optimization of microprogrammed control units. Instruction	8

	cycle, type of instructions, types of operands, Instruction set, addressing modes. Instruction pipelining. Case study of 80386.	
5	**Memory and Input/output Organization** Memory devices and organization, Cache memory organization, principles, memory mapping, basics of virtual memory and address translation, performance characteristics, External memory devices and organization, Introduction to buses, bus organization, DMA organization. Case study of 80386.	8
	Total Hours	**40**

Text Books

1. R. P. Jain, 'Modern Digital Electronics', Tata McGraw-Hill, (5th Edition), (2022)
2. C. Hamacher, Z. Vranesic and S. Zaky, 'Computer Organization and Embedded Systems', McGrawHill, (5th Edition), (2017) W. Stallings, 'Computer Organization and Architecture - Designing for Performance', Prentice Hall of India, (10th Edition), (2016)
3. John P. Hayes, "Computer Architecture & Organization",
4. Douglas Hall, "Microprocessors & Interfacing", McGraw Hill, Revised 2 Edition, 2006 ISBN 0-07-100462-9
5. Intel 80386 Hardware Reference Manual 1986, Intel Corporation, Order no.: 231732-001, 1986.
6. A.Ray, K.Bhurchandi, "Advanced Microprocessors and peripherals: Arch, Programming & Interfacing", Tata McGraw Hill, 2004 ISBN 0-07-463841-6

Reference Books

1. John Yarbrough, "Digital Logic applications and Design", Cengage Learning, ISBN – 13: 978-81-315-0058-3
2. D. Leach, Malvino, Saha, "Digital Principles and Applications"Ⅱ, Tata McGraw Hill, ISBN –13:978-0-07-014170-4.
3. Anil Maini, "Digital Electronics: Principles and Integrated Circuits"Ⅱ, Wiley India Ltd, ISBN:978-81-265-1466-3.
4. Norman B and Bradley, "Digital Logic Design Principles", Wiley, ISBN:978-81-265-1258
5. Brey, Barry B, "8086/8088, 80286, 80386 and 80486 Assembly Language Programming", Prentice Hall, ISBN: 13: 9780023142475.

Online References

1. NPTEL series – nptel.ac.in/courses/117105080/ (Digital System Design by Prof. D. Roychoudhary, Dept. of Computer Science and Engineering, IIT Kh.)
2. Online Chapters – WilliamStallings.com/COA/COA8e.html

Second Year B.Tech Computer Engineering

Semester-III

Course Code: CE24PCC205 **Course Name: Data Structures Laboratory**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	2	-	-	-	-	-	-	-	50	50	-	1	-	1

Prerequisite: Fundamentals of Programming Languages Tutorial (CE24PCC102), Engineering Exploration Lab -1 (CE24VSE103)

Course Objectives:

The course aims to:

- Apply linear data structures to perform insertion, deletion and traversal operations.
- Implement sorting algorithms and hashing techniques to efficiently store and retrieve data.
- Apply tree and graph algorithms to solve real-world problems for data analysis.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Implement linear data structures, searching, sorting, and hashing techniques to solve real-life problems.

CO2: Design solution for real life problems using stack and queue data structures.

CO3: Develop programs using nonlinear data structures for solving computing problems.

List of Experiments

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	**Rainfall Tracking:** Write a program to track rainfall data for 3 cities over 4 months. Using a 2D array, we can store the data, calculate the average rainfall for each city, and display the rainfall data in a tabular format. **OR** **Temperature Tracker:** Write a program for tracking daily temperatures of 3 cities for a week . The program calculates the average temperature for each day and for the week.	4
2	**Game Development:** write a game development program that implements the Bubble Sort algorithm. The program will simulate a simple game where the player can input a set of numbers, and the numbers will be sorted using Bubble Sort to simulate a "level-up" scenario where the player's scores are sorted in ascending order. **OR** **Organizing Cards in a Hand:** Application: When playing card games, players often use an approach similar to insertion sort to organize their cards. They pick one card at a time and insert it into the correct position in their hand, maintaining a sorted sequence. Write a program that demonstrates how to organize (sort) cards in a hand using insertion sort	4
3	**Load Balancing:** For example, imagine you have a set of servers that handle requests for a web application. The key to load balancing is using the hash value of a client's IP address or a request ID to determine which server should handle the request.	2

	The hash function is typically designed so that the data is evenly distributed across the servers, ensuring that no single server is overloaded. Write a program of a load balancing system using hashing, where a basic hash table for mapping incoming requests to a set of servers.	
4	**Simple Task Scheduler:** Write a program that implements a simple task scheduler using a singly linked list. Each node in the linked list represents a task with its priority and execution time. Tasks are scheduled based on their priority, with higher priority tasks being executed first. Note: Use an array to store the URLs. Use a singly linked list to maintain the order of visits.When a new URL is visited: If the history is not full, add it to the head of the list and the array. If the history is full, remove the last node from the list (oldest visit) and replace it with the new URL. Update the head pointer and array index accordingly.	4
5	**Parenthesis Checker:** Write a program using a stack for push, pop, peek, and isEmpty operations. Write isBalanced() Function that Iterates through the input expression, Pushes opening brackets onto the stack. For closing brackets, it checks the top of the stack for a matching opening bracket. Ensures that all opening brackets are matched by the end of the traversal. Main Function: Accepts a string expression from the user. Uses isBalanced() to determine if the parentheses in the expression are balanced. OR **Syntax Parsing in Programming Languages:** Parsing expressions is a key step in many compilers and language processors. When a language's syntax requires parsing mathematical or logical expressions, converting between infix and postfix notation ensures that expressions are evaluated correctly. Accept an infix expression and show the expression in postfix form.	4
6	**Coffee Shop Line (Simple Queue):** Arrival: Customers arrive at the coffee shop and stand in line. Order Processing: The first customer in line gets their order taken, and the barista starts making the coffee.Serving: Once the first customer is served, they leave the queue, and the next customer in line moves forward to be served. Write a program to implement a simple queue **OR** **Printer Spooler (Circular Queue):** In a multi-user environment, printers often use a circular queue to manage print jobs. Each print job is added to the queue, and the printer processes them in the order they arrive. Once a print job is completed, it moves out of the queue, and the next job is processed, efficiently managing the flow of print tasks. Implement the Printer Spooler system using a circular queue without using built-in queues.	4
7	**Web Crawling:** **a. Breadth First Search (BFS):** Application:Indexing web pages for search engines. Example: A web crawler uses BFS to visit web pages systematically, starting from a seed URL and exploring links level by level. Nodes represent web pages. Edges represent hyperlinks. BFS ensures that pages at the same "depth" (distance from the starting page) are visited before moving to deeper levels. Write a program to simulate the indexing of web pages for a search engine using a Breadth-First Search (BFS) algorithm.	2

	b. Depth First Search (DFS): Application: Web crawlers use DFS to explore web pages systematically, following links and indexing content for search engines. Write a simple program to index web pages using Depth First Search (DFS). The program should simulate a web graph where pages are represented as nodes and hyperlinks as edges.	
	Total Hours	**24**
Suggested Mini Projects List **Note:** **1. The Mini Project should be implemented in a group of 3-4 students.** **2. Students can select any one from the following mini project titles.** **3. Students can select a mini project title of their own choice.** **4. Implementation can be in C/C++/Java**		
P1	**Postfix Expression Validator:** A postfix expression is valid if it contains the correct number of operands and operators, and operators have sufficient operands to operate on. You can use a stack to validate a postfix expression by counting the operands and operators during traversal. **Steps**: 1. Traverse the postfix expression from left to right. 2. Push operands onto the stack. 3. When an operator is encountered, check if there are at least two operands to perform the operation. 4. If at the end, exactly one operand remains on the stack, the expression is valid.	6
P2	**Social Network Analyzer:** Analyze a social network (e.g., friendships in a class, followers on Twitter) using graph theory. Collect data on social connections. Represent the network as a graph.Nodes will be as individuals, edges as connections. Calculate centrality measures ,i.e. degree, betweenness, closeness to identify influential individuals. Detect communities within the network. Visualize the network using graph visualization tools.	
P3	**Huffman Coding:** Implement Huffman coding for data compression.Create a Huffman tree based on the frequency of characters in a given text. Assign codes to characters based on their positions in the tree. Encode and decode messages using the Huffman codes	
P4	**Library Book Catalog:** Create a simple library book catalog system.Allow users to add new books to the catalog (with title, author, ISBN, etc.).Implement sorting options for the catalog (e.g., sort by title, author, ISBN, publication year).Allow users to search for books based on keywords. Implement efficient sorting algorithms to handle large catalogs.	
P5	**Customer Service Call Center:** Simulate a basic call center system. Use a queue to store incoming customer calls. Simulate call handling by assigning calls to available agents. Track call waiting times and agent idle times. Analyze call center performance and identify potential bottlenecks.	
P6	**Hospital Emergency Room:** Simulate a hospital emergency room patient triage system. Use a priority queue to prioritize patients based on their medical conditions (e.g., critical, urgent, minor).	

	Simulate patient arrival and treatment times. Analyze the impact of different priority queue implementations on patient wait times and overall system efficiency.	
P7	**Undo/Redo Functionality:** Implement basic undo/redo functionality for a text editor or drawing application.Use a singly linked list to store a history of user actions (e.g., typing, deleting, drawing). Allow the user to undo previous actions by traversing the list backwards.Allow the user to redo undone actions by traversing the list forwards.	
P8	**Music Player Search:** Develop a simple music player with a search feature.Create a library of music files with metadata (artist, album, song title).Implement search algorithms to find songs based on artist, album, song title, or keywords in song lyrics	
P9	**Traffic Light Controller:** Simulate a basic traffic light controller for a single intersection. Use a circular queue to store the sequence of traffic light phases (e.g., green for north-south, yellow for north-south, red for north-south, green for east-west, etc.). Cycle through the phases in the queue to control the traffic lights. Optionally, incorporate features like pedestrian crossing signals and adjust the timing of phases based on traffic volume.	
	Total Hours	**30**

Guidelines for Laboratory Conduction:

1. All the assignments on all concepts are mandatory.
2. Assignments on all concepts covered on Linked list, stack , Queue, Tree and Graph are mandatory and should be implemented on coding platforms such as Leetcode, GitHub.
3. Operating System recommended: - 64-bit Open-source Linux or its derivative.
4. Programming tools recommended: - G++/GCC, Eclipse/Geany.

Guidelines for Students:

1. The laboratory assignments are to be submitted by students in the form of a journal.
2. Journal consists of Vision Mission of Institute, Department, certificate, table of contents and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept, algorithm, time complexity, sample input and expected output, conclusion).

Guidelines for Laboratory Assessment:

1. Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of students.
2. Each Laboratory assignment assessment should assign grade/marks based on rubrics with appropriate weightage.
3. Suggested rubrics for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness.

Text Books

1. Sartaj Sahni, Ellis Horowitz, "Fundamentals of Data Structures in C", Orient blackswan, 2nd Edition, 2010, ISBN 81- 7515-257-5.
2. Michael T. Goodrich, Roberto Tamassia, David M. Mount, "Data Structures and Algorithms in C++", ISBN-13 978-8126512607 Publisher WILEY INDIA EDITION, Publication date 1 January 2015.

Reference Books

1. E. Balagurusamy, "Object Oriented Programming with C++", Eighth Edition, ISBN-13 978-9389949186, McGraw Hill, 24 September 2020

2. Yashavant Kanetkar, "Let us C++", BPB Publications (16 September 2020), ISBN-10 : 9388176642, ISBN-13 : 978-9388176644.
3. G.A.V. PAI, "Data Structures and Algorithms, Concepts, Techniques and Applications", Tata McGraw-Hill, Volume1 1st Edition, 2017. ISBN-10: 0070667268/ ISBN-13: 978-0070667266.
4. Richard F. Gilberg& Behrouz A. Forouzan, "Data Structures, Pseudo code Approach with C", Cengage Learning India Edition, 2nd Edition, 2007, ISBN 10: 8131503143 ISBN 13: 9788131503140.
5. Y. Langsam, M. Augenstinand, A. Tannenbaum,"Data Structures using C & C++",Pearson Education India, Second Edition,2015, ISBN 10: 9332549311, ISBN 13: 978-9332549319.

Online References

1. NPTEL Course: Prof. Naveen Garg, IIT Delhi, "Data Structures And Algorithms" (Link: https://nptel.ac.in/courses/106102064)
2. https://www.cs.usfca.edu/~galles/visualization/Algorithms.html
3. https://visualgo.net/en

Second Year B.Tech Computer Engineering

Semester-III

Course Code: CE24PCC206 **Course Name: Object Oriented Programming Laboratory**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	2	-	-	-	-	-	-	-	50	50	-	1	-	1

Prerequisite: Fundamentals of Programming Languages (CE24PCC101)

Course Objectives:

The Course aims to:

- Understand the principles of object-oriented programming (OOP) with core concepts of object-oriented design.
- Acquire proficiency in the syntax and semantics of the C++ programming language.
- Develop OOP programming skills to enhance advanced programming capability.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Apply key concepts of object-oriented programming to solve everyday challenges.
CO2: Design solutions to real life problems using polymorphism concepts.
CO3: Implement generic programming using template and STL in OOP applications.

List of Experiments

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	**Operations on Bank Account** Implement a class to illustrate working of a bank account of a customer that consists of following members Data members: i. Name of the customer ii. Account Type iii. Account Number iv. Available balance amount in the account Member Functions: i. Assign initial values using constructors. ii. Deposit an amount in account iii. Withdraw an amount iv. Display account detail	2
2	**Employee Information System** Develop a program in C++ to create a database of an employee's information system containing the following fields: Name, employee ID, Department, Date of Joining, Contact address, Telephone number etc. Construct the database with suitable member functions to accept and print employee details. Make use of constructor types, destructor, static members, inline function and dynamic memory allocation using operators-new and delete.	4
3	**Educational Institution System** Design an educational Institution that maintains a database of all staff members where the database is segregated into a number of classes having hierarchical relationships. Define all classes with suitable data members and required member functions to build the database and fetch individual details .	2
4	**Designing of Complex Number calculator** Implement a class Complex which represents the Complex Number. Implement the following functions **Using Operator Overloading:**	4

	1. Constructors (Include all constructor types) 2. Overload operator + to add two complex numbers using member function. 3. Overload operator * to multiply two complex numbers using friend function. 4. Overload operators << and >> to output and accept Complex Numbers	
5	**Geometrical Shapes Area calculator** Define the base class Shape with two data members of any numeric type that are employed to compute the area of respective shape. Add member functions to input the data members values and display the area in base class only. Derive two classes from base class namely, Triangle & Rectangle. Take the display function as virtual in base class and redefine it in the derived classes. Using above classes , write a program to accept parameters for triangle or rectangle and display the area **Using Virtual Functions.**	2
6	**Exception Handling** Design a program with the following functionalities: 1. A function to read two numbers(Double type) 2. A function to calculate division of these two numbers 3. A try block to detect and throw an exception when divide by zero condition occurs 4. Appropriate catch block to handle the exception thrown	2
7	**Sequential Files Operations** Develop a system that defines inventory class with suitable data members and member functions. Perform read and write operations using class objects.	4
8	**Array operations** Design a program with a template for sorting the accepted array and displaying it using integer or float type data. Implement any sorting type **using Generic Programming** .	2
9	**Associative Container Map** Implement a map associative container in which the keys will be the Permanent Registration Number (PRN) of the students and the values will be the name of the respective student. Develop a program that prompts the user for PRN and then it looks in the map, using the PRN as an index and returns the name of the student.	2
	Total Hours	**24**

Suggested Mini Projects List

Note:

1. **The Mini Project should be implemented in a group of 3-4 students.**
2. **Students can select any one from the following mini project titles or can select mini project title of their own.**
3. **MIni project implementation should be preferably in C++.**
4. **Design and implement a mini-project for any application or use case using maximum features of Object Oriented Programming. Use of sequential files is mandatory.**

P1	**Fitness Tracker** Build an application to track workouts and daily activities where users can log their exercises, durations, and calories burned.Classes: Exercise, User, WorkoutPlan	6
P2	**Quiz Application:** Implement a quiz application where users can take quizzes, track scores, and see their performance.	

P3	**Library Management System:** Create a library management system that allows users to add, update, delete, and search for books. Implement functionalities like user authentication, book categorisation, and transaction tracking.	
P4	**Medical Information System** A Medical Information System is designed to store and organize patient medical information, such as demographic and clinical data, lab results, and other relevant information using a C++ programming language. The system is designed to allow easy expansion, and efficient and secure payments using object-oriented programming techniques to ensure the data is stored, retrieved, and managed efficiently. The system can use encryption and authentication techniques to ensure the data is kept secure and private for a particular patient	
P5	**Weather application** Design weather report system that holds a daily weather report with data members dayofmonth,high temp , low temp ,amount rain and amount snow. Display weather report using analysis of data.	
P6	**Task Management Tool** Develop a tool for managing lists of tasks, allowing users to add, edit, and delete tasks. Implement features like setting priorities, due dates, and categorization of tasks.	
P7	**Payroll System** Develop a system that offers a robust payroll application containing features such as salary calculations, deductions, and benefits for both employees and employers. The system facilitates the management of employee records, monitors hours worked, and generates comprehensive report.	
P8	**Stock Management System** Create a Stock Management System assisting businesses in effectively managing their inventory. It monitors current stock levels and provides functionalities for users to add, remove, and modify items within the inventory.	
	Total Hours	**30**

Guidelines for Laboratory Conduction:

1. Assignments on all concepts are mandatory.
2. Assignments should be implemented on coding platforms such as GitHub, HackerRank, CodeChef.
3. Use of open-source software is encouraged.
4. Operating System recommended: - 64-bit Open-source Linux or its derivative.
5. Programming tools recommended: - G++/GCC, Eclipse.

Guidelines for Students:

1. The laboratory assignments are to be submitted by students in the form of a journal.
2. Journal consists of prologue, certificate, table of contents and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, Theory- Concept, algorithm , program code with suitable input and output, conclusion).

Guidelines for Laboratory Assessment:

1. Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of students.
2. Each Laboratory assignment should be evaluated based on parameters with appropriate weightage.
3. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness

Text Books

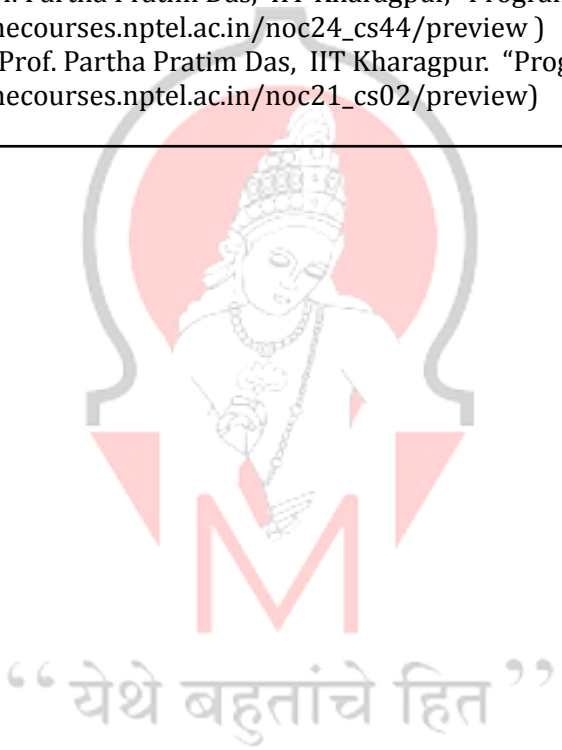
1. E. Balagurusamy, "Object -Oriented Programming with C++", McGraw Hill Education, Eighth Edition, Sept. 2020, ISBN-13 : 978-9389949186.
2. Robert Lafore, "OOP in C++", Pearson Publishing, 4th Edition, 2001, ISBN:0672323087 (ISBN 13: 9780672323089).
3. Yashwant Kanetkar, "Let Us C++", BPB Publications, ISBN: 9789388176644

Reference Books

1. Bjarne Stroustrup, The C++ Programming language, Third edition, 2008, Pearson Education. ISBN 9780201889543.
2. Deitel, C++ How to Program, 4th Edition, Pearson Education,2002, ISBN:81-297-0276-2.
3. Herbert Schildt, C++ The complete reference, Eighth Edition, McGraw Hill Professional, 2011, ISBN:978-00- 72226805

Online References

1. NPTEL Course: Prof. Partha Pratim Das, IIT Kharagpur, "Programming in Modern C++" (Link: https://onlinecourses.nptel.ac.in/noc24_cs44/preview)
2. NPTEL Course: By Prof. Partha Pratim Das, IIT Kharagpur. "Programming in C++" (Link: https://onlinecourses.nptel.ac.in/noc21_cs02/preview)



Second Year B.Tech Computer Engineering

Semester- III

Course Code: CE24CEP208 **Course Name: Project based Learning**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	2

Course Objectives:

The Course aims to:

- Recognize and critically analyze environmental challenges encountered by local communities.
- Explore and evaluate the role of technology in addressing societal needs.
- Design and implement innovative solutions grounded in engineering principles to solve real-world problems.
- Collaborate effectively in teams while contributing individually to the planning and development of projects.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Recall and explain relevant field knowledge, methods, and tools.

CO2: Apply appropriate field research techniques to gather, analyze, and interpret data.

CO3: Analyze field data critically to identify patterns, trends, and challenges, and evaluate their significance.

CO4: Create innovative and sustainable solutions to address complex field-based problems, considering ethical and social implications.

List of Lab Activities

Sr. No	Activities	Duration (Hrs)
1	**Activity 1: Orientation of Field Project based Learning Activities** **Objective:** Familiarize students with the field-based project, clarifying project goals, roles, tasks, and expectations. **Tasks:** 1. PBL Introduction - Brief overview of project-based learning. 2. Project Overview - Outline project scope, objectives, and activities. 3. Team Formation - Assign roles and responsibilities. 4. Timeline & Milestones - Present deadlines and key deliverables. 5. Fieldwork Prep - Discuss tools and methods for data collection. 6. Assessment Criteria - Explain evaluation standards. **7.** Q&A - Address student queries**.** **Expected Outcomes:** 1. Clear understanding of PBL and project scope. 2. Defined team roles and responsibilities. 3. Preparedness for fieldwork. 4. Effective team collaboration. **5.** Understanding of project timeline and deliverables**.**	2
2	**Activity 2:Field Visit & Problem Identification** **Objective:** Evaluate the feasibility of the identified idea/concept from multiple	**4**

	aspects (technical, commercial, operational, and scheduling). **Tasks:** ● Students are expected to visit fields /company/organization to study specific Problem statements. ● Analyze the technical feasibility of the concept (requirements, challenges, technologies involved). ● Assess the commercial feasibility (market demand, potential revenue, cost analysis). ● Examine the operational feasibility (resources, processes, manpower). ● Identify scheduling feasibility (timeframes, project milestones, deadlines). **Expected Outcome:** A comprehensive feasibility report that details the strengths, weaknesses, opportunities, and threats related to the concept.	
3	**Activity 3: Information Gathering, Requirement Analysis, and Literature Survey** **Objective:** Collect relevant information, analyze system requirements, and review existing literature to support the idea. **Tasks :** ● Gather primary and secondary data on the concept and its potential solutions. ● Conduct a requirement analysis for the system (functional, non-functional, and user requirements). ● Perform a literature review of similar projects or systems, identifying key findings and gaps. **Expected Outcome:** A detailed collection of information, a requirements document, and insights from the literature that inform the system's design and development.	8
4	**Activity 4: Review 1** Objective: Review the work done so far to assess the quality and completeness of the gathered information and analysis. **Tasks** : ● Present the results of the feasibility evaluation and information gathering phase. ● Provide an overview of the requirement analysis and the literature survey. ● Receive feedback from peers or mentors on the direction of the project. **Expected Outcome:** Constructive feedback that can be used to refine the problem definition, research, and approach for the next stages of the project.	2
5	**Activity 5: Preparing a System Architecture of the Proposed Idea** **Objective:** Design a high-level architecture for the proposed system that outlines its components and interactions. **Tasks:** ● Define the system's key modules and how they interact. ● Create diagrams representing the architecture (e.g., block diagrams, data flow diagrams). ● Consider scalability, security, and integration with existing systems or infrastructure. **Expected Outcome:** A system architecture document that clearly defines how the system will function and how its components interact with each other.	8
6	**Activity 6: Designing the Proposed System and Specifying Hardware and Software Requirements** **Objective:** Develop the detailed design of the system, specifying the hardware and software requirements. **Tasks:** ● Design system components in detail, including database schemas, interfaces, and logic.	8

	• Identify the hardware specifications (e.g., servers, network devices) and software requirements (e.g., programming languages, frameworks). • Evaluate any dependencies or external systems needed for the implementation. **Expected Outcome:** A design document that includes all hardware and software requirements, detailed designs for each component, and any integration needs.	
7	**Activity 7: Implementation, Preparing Presentation, and Writing Report** **Objective:** Implement the system design, prepare the project presentation, and document the process in a report. **Tasks:** • Develop the system based on the final design, coding the necessary components using hardware and software specifications. • Create a presentation that highlights the system's features, functionality, and outcomes. • Write a comprehensive project report detailing the design, implementation process, and results. **Expected Outcome:** A working system prototype, a professional presentation for stakeholders, and a final report documenting the project from start to finish.	7
8	**Activity 8: Review 2** **Objective:** Perform a final review of the project to assess the overall success and lessons learned. **Tasks :** • Present the completed system, report, and presentation. • Evaluate the system's performance against the original objectives and requirements. • Discuss any challenges faced during the implementation and provide recommendations for future improvements. **Expected Outcome:** A final assessment of the system's effectiveness, a reflection on the project process, and recommendations for improvements or future work.	2
9	**Activity 9: PBL Project Competition** **Objective:** To showcase projects in the best possible way by setting up an engaging display, demonstrating its features, explaining our work clearly, answering questions confidently, and collecting feedback to improve solutions. **Tasks :** • Set up laptop, hardware, models, posters, charts, and other display materials. • Show key features, functionalities, and results to judges and visitors. • Respond confidently to queries from judges, faculty, and visitors. • Deliver a clear, time-bound presentation covering the problem, solution, methodology, results, and impact. **Expected Outcome:** A final assessment of the system's effectiveness, a reflection on the project process, and recommendations for improvements or future work.	4
	Total Hours	**45**
Sample Field Project Based Learning (PBL) Topics		
P1	**Software Development** • **Smart Home Automation System** Design a system that enables users to control and automate various home	8

	appliances remotely from any device, improving convenience, efficiency, and energy management. ● **Task Management Application** Develop an application to help small teams organize and manage tasks, projects, and deadlines efficiently, with features like to-do lists, calendar integration, and notifications. ● **E-commerce Website** Build a platform that enables businesses to sell products online, providing an intuitive interface for product management, customer interaction, and secure transactions. ● **Automated Resume Screening System** Create an application that automatically processes resumes to filter and rank candidates based on their qualifications and suitability for specific job positions.	
P2	**Embedded Systems** ● **Smart Traffic Management System** Develop a system that optimizes traffic flow by detecting traffic conditions and adjusting signals dynamically to reduce congestion and improve traffic efficiency. ● **Weather Monitoring System** Create a system that collects environmental data (e.g., temperature, humidity, pressure) and provides real-time analysis to track weather conditions. ● **Health Monitoring Wearable** Design a wearable device that tracks vital health parameters and alerts users to potential health issues, offering remote monitoring features. ● **Home Security System** Build a system that enhances home security by integrating sensors, cameras, and remote monitoring, enabling users to oversee their property from anywhere.	
P3	**Artificial Intelligence & Machine Learning** ● **Face Recognition System** Develop a system that identifies individuals based on facial features, suitable for applications such as security or attendance tracking. ● **Chatbot for Customer Service** Create an intelligent conversational agent capable of handling customer inquiries, resolving issues, and managing support tickets in an efficient manner. ● **Predictive Analytics for Stock Market** Build a system that predicts market trends by analyzing historical financial data, aiming to provide actionable insights for investors. ● **AI-based Personal Assistant** Design an intelligent assistant that uses natural language processing to perform tasks like setting reminders, managing schedules, and providing information based on voice commands.	
P4	**Data Science & Big Data** ● **Social Media Sentiment Analysis** Develop a system that analyzes social media data to determine public sentiment about specific topics, products, or events, helping brands make informed decisions. ● **Predictive Healthcare System** Build a system that analyzes healthcare data to predict potential health issues and recommends preventative measures or treatments.	

	• **Customer Behavior Analysis** Design a tool to analyze consumer behavior, identify patterns, and suggest marketing strategies to improve customer engagement and sales. • **Data Visualization Dashboard** Create a dashboard that visualizes large datasets, presenting key insights in an easily interpretable format for data-driven decision-making.	
P5	**Cybersecurity** • **Intrusion Detection System (IDS)** Develop a system that detects and alerts about unauthorized access or potential cyberattacks in a network to enhance security. • **Password Strength Analyzer** Build a tool that assesses the strength of passwords and provides recommendations to users for creating stronger and more secure credentials. • **Secure File Sharing System** Design a system for securely sharing files, ensuring privacy through encryption, access control, and audit logs. • **Phishing Detection Tool** Create a system that detects phishing attempts through emails or websites, helping users avoid security risks and fraud.	
P6	**Networking** • **Network Traffic Analyzer** Develop a tool that monitors network traffic, identifies anomalies, and provides insights to optimize performance and detect potential issues. • **VPN System for Secure Browsing** Build a system that ensures secure and anonymous internet access by encrypting user traffic and masking their IP address. • **Wi-Fi Coverage Mapper** Design a tool that maps Wi-Fi signal strength across a given area and provides recommendations for improving signal coverage and connectivity. • **IoT-based Smart Farming System** Develop a system that uses sensor data to monitor agricultural conditions (e.g., soil moisture, temperature) and improve farming efficiency through remote management.	
P7	**Cloud Computing** • **Cloud-based File Storage System** Create a scalable, reliable cloud platform that allows users to store, synchronize, and share files securely across multiple devices. • **Cloud Resource Management Tool** Develop a system for monitoring and optimizing the use of cloud resources, such as virtual machines, storage, and networks, to improve efficiency and reduce costs. • **Serverless Application Architecture** Design an application architecture that eliminates the need for server management by utilizing serverless computing for scaling and cost efficiency. • **Disaster Recovery Plan for Cloud Infrastructure** Create a solution to ensure business continuity by implementing cloud-based disaster recovery strategies, including backup and failover mechanisms.	
P8	**Robotics** • **Autonomous Delivery Robot** Build a robot that can autonomously navigate and deliver items within a designated area, improving efficiency in logistics and transportation.	

	• **Robot Arm for Precision Tasks** Design a robotic arm capable of performing tasks that require high precision, such as assembly, sorting, or packaging, in industrial or research settings. • **Maze Solving Robot** Develop a robot capable of autonomously solving mazes using algorithms, demonstrating problem-solving and decision-making capabilities. • **Robotic Hand for Prosthetics** Create a functional robotic hand prototype that can be used as a prosthetic device, utilizing sensors to detect muscle signals for control.	
P9	**Virtual Reality (VR) / Augmented Reality (AR)** • **Virtual Reality Learning Platform** Develop an immersive learning platform that simulates real-world environments for educational purposes, offering interactive and engaging experiences. • **Augmented Reality Navigation App** Design an augmented reality application that provides real-time navigation assistance in indoor spaces like malls, airports, or museums. • **VR-based Training Simulator** Create a training simulator using virtual reality to provide hands-on, interactive training for fields such as healthcare, engineering, or aviation. • **Interactive AR Shopping Experience** Build an augmented reality app that allows users to visualize how products (such as furniture or clothing) would look in their own environment before purchase.	
P10	**Human-Computer Interaction (HCI)** • **Gesture Recognition System** Develop a system that enables users to control devices through hand or body gestures, offering a more intuitive method of interaction. • **Voice-controlled System for Disabled Users** Create a system that allows individuals with disabilities to interact with computers or devices using voice commands, improving accessibility. • **Eye-tracking Software** Design software that tracks eye movement to control devices or assist with reading, writing, or other activities, offering improved accessibility for users with disabilities. • **Interactive Touchscreen for Education** Build an educational tool that utilizes interactive touchscreens to engage students in learning activities, making education more interactive and dynamic.	
	Total Hours	**53**

Guidelines For Faculty:-Assessment Method

1. Project Milestones: Assessment based on the completion of project milestones (problem definition, design, coding, testing, deployment).
2. Lab Reports: Regular lab activity reports documenting the process, challenges, and solutions.
3. Final Project Presentation: A comprehensive project presentation at the end of the semester.
4. Peer Evaluation: Self-assessment and peer feedback for teamwork and collaboration.
5. Final Project Report: A detailed written report summarizing the entire project lifecycle, including design, implementation, and testing.

Guidelines For Student

Autonomous Learning: Students must learn to initiate tasks and generate ideas, not just imitate. They should think critically, define and analyze problems, and apply their learning to find solutions. They are responsible for their own learning.

Active Participation: Students in PBL are expected to actively construct their knowledge through inquiry. They should not passively receive instruction but engage in the learning process by generating questions, analyzing information, and constructing solutions.
Group Work and Interpersonal Skills: PBL requires students to work in groups, developing interpersonal skills such as effective listening, conflict resolution, and collaborative decision-making.
Inquiry Skills: Students need to critically assess information by asking key questions such as:

- How effective is the solution?
- How strong is the evidence?
- What justifies the chosen method?
- What is the supporting evidence?

Information Literacy: Essential for self-directed learning, this involves identifying when information is needed, locating relevant sources, and using that information to solve problems.
Collaborative Learning: PBL encourages students to work together to solve problems, share ideas, debate, and question each other's perspectives, enhancing learning through collaboration.
Interpersonal Skills for Group Success: Students should be aware of the need for skills like consensual decision-making, dialogue, and discussion to ensure effective problem-solving in group settings.
Note:

1. Students will be given the flexibility to choose their own problem statement.
2. A project basket containing problem statements is available to students.

Text Books

1. Problem Based Learning. byMahnazmoallem, woei hung and Nada Dabbagh, Wiley Publishers. 2019
2. Project-Based Learning in Computer Science" Springer,2020 by Brian H. Kessler "Software Engineering: A Practitioner's Approach" McGraw-Hill Education,9th Edition (2014)by Roger S. Pressman
3. "Database System Concepts" McGraw-Hill Education 7th Edition, 2019 by Abraham Silberschatz
4. "Git Pocket Guide" O'Reilly Media ,2013 by Richard E. Silverman
5. Stem Project based learning and integrated science, Technology, Engineering and mathematics approach. By Robert RobartCapraro, Mary MargaretCapraro

Reference Books

1. De Graaff E, Kolmos A., red.: Management of change: Implementation of problem-based and project-based learning in engineering. Rotterdam: Sense Publishers. 2007.
2. Project management core textbook, 2 Indian Edition , by Gopalan.
3. The Art of Agile Development. By James Shore & Shane Warden.

Online References

1. **PBLWorks:** https://www.pblworks.org/
2. **Edutopia:** https://www.edutopia.org/
3. **GitHub:** https://github.com/
4. **HackerRank:** https://www.hackerrank.com/
5. **LeetCode:** https://leetcode.com/
6. **Kaggle:** https://www.kaggle.com/

Second Year B.Tech Computer Engineering

Semester- III

Course Code: CE24EEM209 Course Name: Human Values and Professional Ethics

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	-	2	-	-	-	-	50	-	-	50	-	-	2	2

Course Objectives:

The course aims :

- To promote ethics,human values, honesty and professionalism.
- To set standards that are expected to follow and to be aware that If one acts unethically what are the consequences.
- To provide basic knowledge about engineering Ethics, Variety of moral issues and Moral dilemmas, Professional Ideals and Virtues.
- To provide basic familiarity about Engineers as responsible Experimenters, Research Ethics, Codes of Ethics, Industrial Standards, Exposure to Safety and Risk, Risk Benefit Analysis

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand the basic perception of human values and professional ethics.

CO2: Demonstrate the impact of engineering ethics in societal and environmental contexts.

CO3: Apply professional ethics and workplace responsibilities to ensure safety and ethical engineering.

CO4: Apply ethical principles to resolve global issues and situations that arise in their professional lives.

Contents

1	**Human Values:** Morals, Values, and Ethics; Integrity; Work Ethic; Service Learning; Civic Virtue Respect for Others, Living Peacefully, Caring, Sharing, Honesty, Courage, Cooperation, Commitment, Empathy, Self-Confidence. Case Study- Selflessness in Crisis: The Story of a COVID-19 Volunteer.
2	**Engineering Ethics:** Senses of 'Engineering Ethics' Variety of moral issues, Types of Inquiry, Moral dilemmas, Moral Autonomy Kohlberg's theory, Gilligan's theory, Consensus and controversy, Models of professional roles. Case Study-The Space Shuttle Challenger Disaster: Engineering Failures and Ethical Oversight
3	**Ethics in Social Media and Technology:** Safety and risk— Assessment of safety and risk in social media, Navigating moral values related to responsible online behaviour, transparency,authenticity, respect, and data privacy, Risk- benefit analysis and reducing risk. Case Study: AI-Driven Undercover Surveillance: The Overwatch Program Workplace Responsibilities and Rights- Collegiality and loyalty - respect for authority confidentiality, conflicts of interest, occupational crime, professional and employee rights Case Study : Whistleblowing in the Workplace: Protecting Employees' Rights and Ethical Responsibilities

4	**Global Challenges and Environment Responsibilities:** Globalization, Cross cultural issues, Multinational corporations, Environmental ethics, weapons development, Engineers as managers consulting engineers. Case study- Carbon Footprints in the Corporate World: How Companies Can Lead the Fight Against Climate Change

List of Tutorial (Any 8)

Sr. No.	Name of the Tutorial	Duration (Hrs.)
1	Purpose-Introduce the concept of Professional Code of Conduct Method – Using Group Discussion as a platform, ask students to share one practice in their family / home that everyone has to follow. For ex. not wearing footwear in the house, taking a bath first thing in the morning, seeking blessings from elders, etc. Connect this Code of Conduct in their family to one that exists in the professional world Outcome – Awareness of profession-specific code of conduct and importance of adherence of that code specified. Ability to express opinions verbally and be empathetic to diverse backgrounds and values	4
2	Purpose-Highlight the importance of professional ideals like conflict management, ambition, ethical manners and accountability Method – Each student will have to write a 200 word essay on any of above mentioned virtues of being a good professional. On evaluation, the top 5 essays can be displayed on the college wall magazine and rewarded if deemed appropriate Outcome – Learn to express one's ideas and identify and relate to good virtues. Build writing skills, improve language and gain knowledge about how to write an impactful essay	2
3	Purpose – Make students aware that technology can be harmful if not used wisely and ethically Method – Conduct a quiz on various ethical dilemmas that are relevant in today's world pertaining to privacy right, stalking, plagiarism, hacking, weaponizing technology, AI, electronic garbage creating environmental hazard Outcome – Make students aware of various adverse consequences of technology development and allow them to introspect on how to use technology responsibly.	8
4	Purpose – Provide an insight into rights and ethical behavior. Method – Movies like The Social Network can be played and students can be asked to discuss their opinion about collegiality, intellectual property, friendship and professional relationships Outcome – help them look at success stories from an ethical point of view. Develop critical thinking and evaluation of circumstances.	4
5	Purpose – Expose students to professional situations where engineers must use their skills ethically and for the betterment of society and nation Method – Students in groups of 5 can be given an assignment in the earlier session to present in front of the class one specific case where they felt unethical treatment has been meted out to a person by an engineer – either as a witness, advisor, dishonesty, improper skills testimony etc. The group had to make a short presentation and also suggested possible solutions to that situation. Q&A from other students must encouraged to allow healthy discussion	4

	Outcome – Become aware of unethical code of conduct in the professional world and how to follow a moral compass especially when one reaches positions of power.	
6	Purpose-Impress upon the students, the significance of morality Method – Role plays a professional situation where an engineer is not competent and is trying to copy the work of a colleague and claim credit for that work. Ask observing students to react to that situation. Alternatively, a short video that clearly shows unethical behavior can be played and ask viewers their opinion about the situation. Note to teachers – read about Kohlberg's theory and Gilligan's theory to understand levels of moral behavior Outcome – Incite students to contemplate their own immoral behavior in public space or academic environment (like copying homework or assignment). Will coax students to introspect their own values and encourage them to choose the right path	4
7	Purpose-Make students aware of proper and globally accepted ethical way to handlework,colleagues and clients Method – Teacher can form groups of 4-5 students and assign them different cases (these can be accessed online from copyright free websites of B-school content) Outcome – Develop group communication skills. Learn to speak up one's opinion in a forum.Cultivate the habit of presenting solution-driven analytical arguments making them contributors in any team.	2
8	Purpose – Make students contemplate about ideal and safe professional environment and decide on making right decisions based on codes of conduct Method – Students can be asked to write down 5 most important codes of conduct that they feel that every computer engineer should follow. After evaluation by teachers / experts, the collection Outcome – Introspection and think about how to shape the professional environment. Also, when they carry back with them their own codes of conduct, they could feel bound to adhere to these ethics.	4
	Total Hours	**26**

Text Books

1. Mike Martin and Roland Schinzinger, "Ethics in Engineering", McGraw-Hill, 4th edition 2019 New York.
2. Govindarajan M, Natarajan S, Senthil Kumar V. S, "Engineering Ethics", Prentice Hall of India, 3rd Edition 2020,New Delhi.

Reference Books

1. Charles D. Fleddermann, "Engineering Ethics", Pearson Education Prentice Hall, 4 th Edition 2022 New Jersey.
2. 2. Charles E Harris, Michael S. Protchard and Michael J Rabins, "Engineering Ethics – Concepts and Cases",Wadsworth Thompson Learning, 6th Edition 2019 United States.

Online References

1. NPTEL Course: Prof. A.K. Sharma, IIT Kanpur, "Exploring Human Values: Visions of Happiness and Perfect Society" ([Link :https://nptel.ac.in/courses/109104068)
2. NPTEL Course: Prof. Susmita Mukhopadhyay, IIT Kharagpur , "Ethics in Engineering Practice" (Link: https://swayam.gov.in/nd1_noc20_mg44/preview)

Second Year B.Tech Computer Engineering

Semester- III

Course Code: SH24VEC201 **Course Name: Environment Sustainability**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	-	2	-	-	-	-	50	-	-	50	-	-	2	2

Prerequisites: A basic understanding of environmental science, geography, or social studies

Course Objectives:

The course aims to:

- Understand the foundational concepts of sustainability, including its social, environmental, and economic dimensions, and the global need for sustainable development.
- Gain insights into the 17 United Nations Sustainable Development Goals (SDGs) and the role of technology, international cooperation, and policy in achieving them.
- Identify and explain different types of pollution (air, water, solid waste), their sources and impacts, and sustainable approaches to waste management including the 3Rs (Reduce, Reuse, Recycle) and zero-waste concepts.
- Evaluate global and local environmental challenges, including climate change, global warming, ozone depletion, and resource degradation

Course Outcomes:

After completion of the course, the students can:

CO1: Describe the fundamental ideas of environmental sustainability.

CO2: Examine sustainability's social, economic, and environmental facets.

CO3: Analyse how human actions affect Environmental systems.

CO4: Propose strategies for sustainable development in various sectors.

Contents

1	**Introduction to Environmental Sustainability:** Global environmental issues, Foundations of Environmental Sustainability. Sustainability: Challenges, potential; role of technology, policies.
2	**Environmental Pollution and Waste Management:** Air Pollution: Sources and Effects of Air Pollution. Water Pollution: Sources of Water Pollution, Sustainable Wastewater Treatment. Solid Waste zero Waste Concepts (3R Concept).
3	**Climate Change and Energy:** Understanding Climate Change, Science and impacts of climate change Climate modeling and adaptation, Sustainable Energy Solutions, Renewable energy technology, Efficient energy system and conservation
4	**Sustainable Development Practices:** Sustainability in Cities, infrastructure and cities that are sustainable, green buildings and smart cities.

List of Tutorial (Any 8)

Sr. No.	Name of the Tutorial	Duration (Hrs.)
1	Examine the river pollution in Pune.	4
2	Prepare a climate change strategy for your town, city, or building.	4

3	Initiate a campaign to bring emphasis to a sustainability concern.	4
4	Examine The Current Environmental Issue	2
5	Examine a successful sustainability attempt from the real world	2
6	Home energy audit: Have students make a list of all the appliances and light bulbs in their house. How much energy does their house use if all the lights are on for 4 hours per day? If their appliances are on for 2 hours per day? How much energy could they save if they switched to energy-efficient appliances or lightbulbs?	2
7	Use recycled material in art projects: recycled materials can make beautiful art projects such as jewelry, planters, and birdhouses. Incorporating materials that would otherwise be thrown away into art projects can show your students how to find new uses for these items.	2
8	Life cycle: One way to show students what happens when you put something in the trash versus recycling or reusing the object is to do a life cycle analysis. This is a flowchart that shows the environmental impacts of an object, from extracting the raw materials to decomposition and everything in between. When something is put in the trash instead of being reused or recycled, the life cycle assessment will show a bigger environmental impact. When something is reused or recycled, the environmental impact is less because raw materials don't need to be extracted to create something new.	2
9	Go on a field trip: Visit your local landfill, recycling center, or a nearby composting facility where the students can see firsthand what is happening to waste and learn about the lifecycle of waste and its effect on the environment.	2
10	Compare carbon footprints of different materials used in construction or manufacturing in India.	2
	Total Hours for 08 Activities	**26**

Text Books

1. Sharma, "P.D. Ecology and Environment", Rastogi Publications,2020.
2. Kaushik, C.P., Kaushik, Anubha, "Perspectives in Environmental Studies New Age",International Publishers, 2021.
3. Agarwal, S.K. "Environmental Management", APH Publishing Corporation, 2005.

Reference Books

1. Kalam, A.P.J. Abdul, Srijan Pal Singh, "Target 3 Billion Innovative Solutions Towards Sustainable Development", Penguin Books,2011.
2. Kolbert, Elizabeth, "The Sixth Extinction: An Unnatural History", Henry Holt and Co.,2014.

Online References

1. NPTEL Course : Prof. Trupti Mishra, IIT Bombay "Business and Sustainable Development" [https://nptel.ac.in/courses/110101153]
2. NPTEL Course :By Prof. Brajesh Kumar Dubey , IIT Kharagpur "Sustainable Engineering Concepts And Life Cycle Analysis" [https://onlinecourses.nptel.ac.in/noc23_ce90/preview]

SEMESTER IV

Second Year B.Tech Computer Engineering

Semester-IV

Course Code: CE24PCC251 **Course Name: Database Management Systems**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3	-	-	-	-	40	60	-	-		100	3		-	3

Prerequisite: CE24MDM204:Mathematical Foundation,CE24PCC201: Data Structures

Course Objectives:

1. To understand database concepts, design principles, and ER/EER modeling.
2. To solve SQL queries and PL/SQL subroutines for efficient data processing.
3. To apply normalization techniques for designing well-structured relational databases
4. To explore database transactions, concurrency control methods, and recovery mechanisms.
5. To analyze NoSQL database models and their role in managing unstructured data.

Course Outcomes:

After learning the course, the students will be able to:

CO1: To understand fundamental of DBMS including data models and database design.
CO2: Develop SQL queries and PL/SQL subroutine to manage and manipulate relational data
CO3: Apply normalization techniques for database design.
CO4: Analyze transaction management concepts and concurrency control techniques for reliable database systems.
CO5: Evaluate NOSQL database models and compare emerging database technology.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Database Management Systems and ER Model** Introduction, Purpose of Database Systems, Database-System Applications, View of Data,Database Languages, Database System Architecture and Structure, Data Models.**Database Design and ER Model:** Entity, Attributes, Relationships, Constraints, Keys, Design Process, ER Diagram, Design Issues, Extended E-R Features, converting ER and EER diagrams into tables. **Case Study:** Study of Architecture of any DBMS like Oracle or MySQL. Design a database schema	8
2	**SQL and PL/SQL** **SQL:** Characteristics and Advantages,SQL Data Types and Literals, **SQL Commands:** DDL, DML, DCL, TCL**,** Select queries, String operations, JOINs,Setoperations,SetMembership,Views,Indexes,Sequences,Aggregate,Nested Subqueries. PL/SQL: Stored Procedures, Functions, Cursors,Triggers. **Case Study :** Design and implement a Student Course Management System using SQL and PL/SQL to manage students, courses, and faculty members efficiently. The system should store and retrieve relevant data, ensuring integrity, security, and performance optimization.	8
3	**Relational Database Design & Normalization** **Relational Model:** Basic concepts, Attributes and Domains, CODD's Rules. **Relational Integrity:** Domain, Referential Integrities, Enterprise Constraints, **Database Design:** Features of Good Relational Designs ,Functional Dependencies,1NF,2NF,3NF,BCNF. **Case study:** Design and Optimization of a Relational Database for a University Management Systemright-linear grammar, Applications of grammar	8

4	**Database Transaction Management and Concurrency Control** Basics of Transaction Management,ACID property, Schedules, Serializability, Locking and timestamp-based schedulers, Concurrency control,multi-version and optimistic Concurrency Control schemes, Database recovery. **Case study:** Design Online Shopping Cart Transaction Management In an e-commerce platform, multiple users simultaneously add, update, and purchase products. To ensure data consistency and reliability, the system must handle concurrent transactions effectively.	8
5	**NoSQL and Emerging Database Technologies** **Distributed Databases:** Features, advantages, disadvantages, CAP theorem,**Types of Data:** Structured, unstructured, Semi-structured, **NoSQL Overview:**Introduction, need, features; ACID vs BASE,**Types of NoSQL Databases**: Key-value, document, graph, wide-column stores. **MongoDB Essentials:** CRUD operations, indexing, aggregation, MapReduce. **Emerging Database Technologies:**XML,Mobile Database **Case study:** Study NoSQL Database Selection for a Social Media Platform	8
	Total Hours	**40**

Text Books

1. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 7th edition
2. Connally T., Begg C., "Database Systems", 4th Edition, Pearson Education, 2002, ISBN 8178088614
3. D TEditorial Services "BIG DATA Black Book", Dreamtech Press ISBN 13 : 978935119931

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",Emereo Pty 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
8. Seema Acharya, "Demystifying NoSQL", Wiley Publications, ISBN: 9788126579969

Online References

e-Books :

- SQL and Relational Theory a. (How to Write Accurate SQL code), C.J. Date, O'REILLY Publication.
- SQL A Beginner's Guide, Andy Oppel, Robert Sheldon, McGraw Hill Publication
- NPTEL Course: Prof. Sreenivasa Kumar, IIT Madras, "Introduction to Database Systems" https://nptel.ac.in/courses/106106220
- NPTEL Course: Prof. Samiran Chattopadhyay,IIT Kharagpur, "Data Base Management System" https://nptel.ac.in/courses/106105175
- Mongodb Course: https://www.mongodb.com/resources/basics/databases/nosql-explained

Second Year B.Tech Computer Engineering

Semester-IV

Course Code: CE24PCC252 **Course Name: Principles of Programming Languages**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	3

Prerequisite: Object Oriented Programming (CE24PCC202)

Course Objectives:

The course aims :

- To learn basic principles of programming languages and programming paradigms.
- To learn advanced Java features including inheritance, packages, interfaces, exception handling, and file input/output for building robust applications.
- To learn the use of Java collections and multithreading to manage data efficiently and handle concurrent execution in program

Course Outcomes:

After learning the course, the students will be able to:

CO1: Analyze different programming languages based on their core principles and features.
CO2: Apply Object-Oriented Programming (OOP) concepts to implement fundamental Java constructs.
CO3: Implement the principles of inheritance, encapsulation, and package usage.
CO4: Implement the concept of exception handling to effectively manage errors during I/O operations, with file handling tasks.
CO5: Analyze the concept of Multithreading for robust application development.

Unit	Contents	Duration (Hrs.)
1	**Fundamentals of Programing** Importance of Studying Programming Languages, History of Programming Languages, Role of Programming Languages, Programming Environments. Binding & Binding Times. **Programming paradigms-** Introduction to programming paradigms, Introduction to four main Programming paradigms- procedural, object oriented, functional, and logic and rule based.	8
2	**Java as Object Oriented Programming Language** **Classes and Methods:** class fundamentals, declaring objects, assigning object reference variables, constructors, this keyword, garbage collection, finalize() method, overloading methods, argument passing, object as parameter, returning objects, access control, static, final, nested and inner classes, command line arguments, variable - length arguments. **Arrays:** one dimensional array, multi-dimensional array, alternative array declaration statements **String Handling:** String class methods	8
3	**Inheritances, Packages and Interfaces** **Inheritances:** member access and inheritance, super class references, Using super, multilevel hierarchy, constructor call sequence, method overriding, dynamic method dispatch, abstract classes, Object class. **Packages and Interfaces:** defining a package, finding packages and CLASSPATH, access protection, importing packages, interfaces (defining,	8

	implementation, nesting, applying), variables in interfaces, extending interfaces, instance of operator.	
4	**Exception handling , Managing I/O & File handling** **Exception handling:** exception types, uncaught exceptions, try, catch, throw, throws, finally, multiple catch clauses, nested try statements, built-in exceptions, custom exceptions (creating your own exception subclasses). **Managing I/O & File handling:** Streams, Byte Streams and Character Streams, Predefined Streams, Reading console Input, Writing Console Output, Printwriter class, File Class, Reading and Writing to Files	8
5	**Collections & Multithreading in JAVA** **Collections:** Collection Framework- List, Set, Map , Queue. **Concurrency and Synchronization, Java Thread Model:** Thread priorities, Synchronization,Messaging, Main Thread, Creating thread: Implementing Thread using thread class and Runnable Interface. Creating multiple threads using is Alive() and join().	8
	Total Hours	**40**

Text Books

1. T. W. Pratt, M. V. Zelkowitz, "Programming Languages Design and Implementation, 4th Ed, PHI, ISBN 81-203-2035-2.
2. Sebesta R., "Concepts of Programming Languages", 12th Edition, Pearson Education, ISBN-81-7808-161-X.
3. Herbert Schildt, "The Complete Reference Java", 13th Ed, TMH,ISBN: 978-0-07-180856-9.

Reference Books

1. Dr. R. Nageshwar Rao, "Core Java: An Integrated Approach",8th Ed, Dreamtech Press,ISBN 9789351199250
2. Carlo Ghezzi, Mehdi Jazayeri, "Programming Language Concepts", 3rd Ed,,Wiley Publication ISBN : 978-81-265-1861-6.

Online References

1. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013842053032140800 10965_shared/overview
2. https://spoken-tutorial.org/tutorial-search/?search_foss=Java&search_language=
3. https://www.codechef.com/roadmap/java-dsa
4. NPTEL Course: Prof. Debasis Samanta, IIT Kharagpur, "Programming in Java" https://nptel.ac.in/courses/106105191

Coding Practice:

1. https://www.hackerrank.com/domains/java
2. https://leetcode.com/problemset

Second Year B.Tech Computer Engineering

Semester- IV

Course Code:CE24PCC253 **Course Name: Software Engineering**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
3	-	-	-	-	40	60	-	-	-	100	3	-	-	3

Course Objectives:

The Course aims :

- To learn the fundamental concepts of software engineering, including software development methods, process models, and agile strategies.
- To learn how to identify and analyze user needs and system requirements for developing well-defined software specifications.
- To learn estimation techniques for analyzing software project effort and scheduling.
- To learn how to model software systems structurally using Unified Modeling Language (UML) diagrams.
- To learn the significance of Software Configuration Management (SCM) repositories and evaluate various software testing strategies.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Describe the concepts of software engineering methods, process models and agile Development Strategies.

CO2: Apply user needs and system requirements analysis to develop comprehensive software requirements.

CO3: Analyze project effort/schedule using appropriate estimation techniques.

CO4: Draw Structural models of software systems using UML Diagrams.

CO5: Recall the role of SCM repositories and the effectiveness of test strategies.

Unit	Contents	Duration (Hrs.)
1	**Introduction to Software Engineering** Nature of Software, Software Engineering Layers, Software Development Life Cycle (SDLC) Process models: Framework Activity, Process Pattern, waterfall model, incremental process models, evolutionary process models-Prototyping and Spiral, the unified process. Agile Software Development:Agility, Agile Process, Agile vs. Plan-Driven Development. Agile Methodologies: Introduction of Scrum, Extreme Programming (XP), Adaptive Software Development(ASD), Feature Driven Development(FDD), Introduction of Jira tool.	8
2	**Requirements Engineering** Introduction to Requirements Engineering-Requirements Engineering, Establishing the groundwork,Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Requirements Negotiation and Validating Requirement, requirement analysis, scenario based modelling, requirement modelling strategies, flow oriented modelling, class based modelling, Software Requirements Specification (SRS).	8

3	**Estimation and Scheduling** Software Project Estimation:Importance of Software Project Estimation, Estimation Decomposition Techniques-Software Sizing, Problem-Based Estimation, LOC-Based Estimation, FP-Based Estimation, The COCOMO II Model. Project Scheduling: Project Scheduling, Defining a Task for the Software Project, Scheduling.	6
4	**Design Engineering and Risk Management** Design Engineering-Design process, design concepts, the design model: Data Design, Architectural, Interface design elements. Architectural design-software architecture, architectural styles, architectural design and pattern, conceptual model of UML, basic structural modeling:class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams. Risk Management-Reactive vs Proactive risk strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring, and Management, The RMMM Plan.	9
5	**Software Configuration Management and Testing** Software Configuration Management: Software Configuration Management, The SCM Repository , The SCM Process. Version Control Systems: Git basics and collaboration. Testing Fundamentals:A Strategic Approach to Software Testing-Verification and Validation, Organizing for Software Testing, Software Testing Strategy—The Big Picture,Strategic Issues, Test Strategies for Conventional Software, Unit Testing, Integration Testing, Validation Testing, System Testing.	9
	Total Hours	**40**

Text Books

1. Roger S. Pressman's "Software Engineering: A Practitioner's Approach" 9th Edition, published in 2020 by McGraw Hill.
2. Dan Pilone and Neil Pitman's UML 2.0 in a Nutshell First Edition, published by O'Reilly Media in June 2005
3. Ian Sommerville's "Software Engineering" 10th edition, published by Pearson

Reference Books

4. Bob Hughes, Mike Cotterell, Rajib Mall: Software Project Management, 6th Edition, McGraw Hill Education, 2018.
5. An Integrated Approach to Software Engineering by Pankaj Jalote 3rd Edition, published by Springer in 2005.
6. "Software Engineering: Principles and Practice", Hans van Vliet, Wiley India, 3rd Edition, 2010.
7. Jim Arlow, Ila Neustadt, —UML 2 and the unified process –practical object-oriented analysis and design‖ Addison Wesley, Second edition, 2005, ISBN 978-0201770605
8. Grady Booch et al, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2007

Online References

1. NPTEL Course: Prof. Rajib Mall, IIT Kharagpur, "Software Engineering"(Link)
2. NPTEL Course: Prof. Ramesh Anbanandam, IIT Roorkee, "Project Management"(Link)

Second Year B.Tech Computer Engineering

Semester- IV

Course Code: CE24PCC255 **Course Name: Database Management Systems Laboratory**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	**P**	**T**	**OL**	**ODL**	**CIE**	**ETE**	**TW**	**OR**	**PR**	**TOTAL**	**L**	**P**	**T**	**TOTAL**
-	2	-	-	-	-	-	-	-	50	50	-	1	-	1

Prerequisite: CE24PCC252 - Database Management Systems

Course Objectives:

The course aims to:

- To Understand database concepts and ER/EER modeling.
- Develop SQL and PL/SQL programs for database operations..
- Implement advanced SQL and PL/SQL features like joins, triggers, and cursors.
- Explore NoSQL databases (CRUD, indexing, aggregation, MapReduce).
- Integrate databases with front-end applications for real-world projects.

Course Outcomes :

After completing the course, the students will be able to:

CO1: Apply the concepts of database design by creating E-R diagrams and converting them into relational models.

CO2: Develop SQL queries for data manipulation, transaction control, and access management using DML, DCL, and TCL commands.

CO3: Implement SQL operations, including joins, views, subqueries, stored procedures, and triggers, to optimize data retrieval and integrity.

CO4: Design CRUD operations in MongoDB, demonstrating an understanding of NoSQL Database concepts and their practical applications.

List of Experiments

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	**Case Study of ER diagram** Study the ER (Entity-Relationship) diagram and design ER features using tools such as ERDPlus, ERwin, etc. Convert the ER diagram.	2
2	**SQL DDL** Design and Develop SQL DDL statements which demonstrate the use of SQL objects such as Table, View, Index, Sequence, Synonym	2
3	**SQL DML** Design at least 10 SQL queries for suitable database application using SQL DML statements: Insert, Select, Update, Delete with operators, functions, and set operators.	2
4	**JOINs** Design at least 10 SQL queries for suitable database application using SQL DML statements:all types of Join, Sub-Query and View	2
5	**PL/SQL Stored Procedure and Stored Function.(use control structure and Exception handling)** Write a Stored Procedure namely proc_Grade for the categorization of students. If marks scored by students in examination is <=1500 and marks>=990 then	2

	student will be placed in distinction category if marks scored are between 989 and900 category is first class, if marks 899 and 825 category is Higher Second Class Write a PL/SQL block for using procedure created with above requirement	
6	**Cursors: (All types: Implicit, Explicit, Cursor FOR Loop, Parameterized Cursor)** Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped	2
7	**Database Trigger (All Types: Row level and Statement level triggers, Before and After Triggers).** Write a database trigger on the Library table. The System should keep track of the records that are being updated or deleted. The old value of updated or deleted records should be added in the Library_Audit table.Frame the problem statement for writing Database Triggers of all types, in-line with the above statement. The problem statement should clearly state the requirements.	2
8	**MongoDB** Design and Develop MongoDB Queries using CRUD operations. (Use CRUD operations, SAVE method, logical operators)	2
9	**Aggregation and indexing** Implement aggregation and indexing with suitable examples using MongoDB.	2
10	**Map reduce operation** Implement Map reduce operation with suitable examples using MongoDB.	2
11	**Database Connectivity:** write a program to implement MySQL/Oracle database connectivity with any front end language to implement Database navigation operations (add, delete, edit etc.)	4
	Total Hours	**24**

Suggested Mini Projects List

Note:

1. Design and implement a database application that showcases key DBMS features including ER modeling, SQL (DDL/DML), joins, subqueries, PL/SQL (procedures, functions, cursors, triggers), and front-end connectivity. Develop application considering

1. FrontEnd:Java
2. Backend:MongoDB/MySQL/Oracle

P1	**College Course Registration System** Focus: ER Diagram, SQL DDL & DML, Normalization Design an ER diagram for student, course, department, and enrollment. Convert to relational tables, normalize to 3NF. Create tables using DDL; insert sample data using DML.	
P2	**Online Library Management System** Focus: SQL Views, Indexes, Triggers Create Book, Member, Issue, Return tables. Use views to show books currently issued. Create an index on Book Title. Create Before/After Triggers to log updates/deletes in a Library_Audit table.	
P3	**Student Grading System** Focus: Stored Procedures & Exception Handling Table: Stud_Marks(name, total_marks), Result(Roll, Name, Class) Create a procedure proc_Grade to assign classes based on marks. Call this procedure in a PL/SQL block with error handling.	

P4	**Employee Payroll Management** Focus: SQL DML, Joins, Subqueries Tables: Employee, Department, Payroll, Leaves Use inner, outer, self joins to display employee data. Write subqueries to find top-paid employees, unpaid leaves, etc.	6
P5	**Attendance Tracking System** Focus: Parameterized Cursors Tables: N_RollCall, O_RollCall Use parameterized cursor to merge new roll calls into old, skip duplicates	
P6	**E-commerce Product Inventory** Focus: MongoDB CRUD & Indexing Collections: products, orders, users Perform all CRUD operations using save, update, delete. Apply indexing on product category and price	
P7	**Sales Data Analysis** Focus: MongoDB Aggregation & MapReduce Collection: sales(date, item, qty, price) Use aggregation to calculate monthly sales per item. Use MapReduce to find best-selling product categories.	
P8	**Hospital Patient Management** Focus: Triggers & Stored Functions Tables: Patient, Treatment, Billing, Audit Create a stored function to calculate treatment cost. Triggers to store deleted/updated patient records in Audit.	
P9	**Banking System Interface** Focus: MySQL/Oracle DB Connectivity Build a GUI in Python/Java/PHP to interact with MySQL/Oracle. Perform database navigation: add, edit, delete customer/account data.	
P10	**Movie Ticket Booking System (MongoDB)** Focus: MongoDB + Frontend Connectivity Collections: movies, users, bookings Use a GUI (Node.js, React, or Python Tkinter) to connect to MongoDB. Allow ticket booking, cancellation, and user data update.	
	Total Hours	**30**

Guidelines for Laboratory Conduction:

3. Use Relational DBMS concepts to implement the problem statements.
4. Software Required:
    4. 64 bit open source operating system- Ubuntu
    5. MySQL
    6. Oracle 11g Express Edition(Ubuntu 16.04LTS)
    7. MongoDB
    8. Java

Guidelines for Students:

1. The laboratory assignments are to be submitted by students in the form of a journal.
2. Journal consists of prologue, certificate, table of contents and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept, algorithm, time complexity, sample input and expected output, conclusion).

Guidelines for Laboratory Assessment:

6. Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of students.

7. Each Laboratory assignment should be evaluated based on parameters with appropriate weightage.
8. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness.

Text Books

1. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 7th edition
2. Connally T., Begg C., "Database Systems", 4th Edition, Pearson Education, 2002, ISBN 8178088614
3. D TEditorial Services "BIG DATA Black Book", Dreamtech Press ISBN 13 : 978935119931
4. Murach J., "Oracle SQL and PL/SQL", Murach Publishing, ISBN 978-1943872367, 3rd Edition.

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. Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publications ISBN: 9788176569644, 9788176569644
6. Connolly T., Begg C., "Database Systems: A Practical Approach to Design, Implementation, and Management", Pearson Education, ISBN 978-1292061184, 6th Edition.

Online References

e-Books :

1. Date C. J., "SQL and Relational Theory: How to Write Accurate SQL Code", O'Reilly Media, 2nd Edition.
2. NPTEL Course: Prof. Sreenivasa Kumar, IIT Madras, "Introduction to Database Systems" https://nptel.ac.in/courses/106106220
3. NPTEL Course: Prof. Arnab Bhattacharya, IIT Kanpur, *"Database Management Systems"* https://nptel.ac.in/courses/106104135
4. MongoDB University Course: "MongoDB Basics (CRUD, Aggregation, Indexing)", MongoDB Inc. https://www.mongodb.com/learn/mongodb-basics

Second Year B.Tech Computer Engineering

Semester- IV

Course Code: CE24PCC256 Course Name: Principles of Programming Languages Laboratory

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	2	-	-	-	-	-	-	-	50	50	-	1	-	1

Prerequisite: Object Oriented Programming Laboratory (CE24PCC207)

Course Objectives:

The Course aims **:**

- To learn how to apply object-oriented programming concepts such as classes, objects, inheritance and encapsulation to develop real-world Java applications.
- To learn how to use arrays, collections, and file handling techniques for data storage, manipulation, and retrieval in various problem-solving scenarios.
- To learn to implement exception handling and multithreading in Java to create robust and concurrent applications simulating real-life applications.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Apply object-oriented principles to design and implement Java applications.

CO2: Use Java collections framework efficiently to manage dynamic data sets.

CO3: Implement multithreading and synchronization in Java to handle concurrent tasks in real-world scenarios.

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	**Student Performance Evaluation System.** Create a Student class with attributes like studentID, name, and marks. Write methods to: ● Set and get student information. ● Calculate and display the average marks. ● Determine if the student has passed or failed based on a passing threshold (e.g., 50%). Use Class, objects and Encapsulation	2
2	**Array Puzzle Solver.** Create a Java application that solves different types of puzzles using array manipulation. The system will include a variety of puzzles involving arrays, such as finding missing numbers, sorting and searching problems, finding the largest/smallest elements, sum pair puzzle and other logical tasks based on arrays	4
3	**TextTransformer.** Create a Java application to perform a wide range of string manipulation and transformation operations. The TextTransformer tool will provide users with the ability to process and modify text in various ways, including converting text formats, performing string comparisons, searching and replacing substrings, extracting specific patterns.	2
4	**Vehicle Rental System.** Develop a vehicle rental system where different types of vehicles (e.g., Car, Bike, Truck) inherit from a base Vehicle class. This system allows users to rent vehicles based on type and calculate rental fees.	2

5	**Library Management System.** Develop a simple library system where users can check out and return books. Implement exception handling for cases like books not found, invalid return dates, or when the book is already checked out.	2
6	**Expense Tracker:** 1. Implement the system that manages personal expenses. Record daily expenses, including categories (e.g., food, transportation). Display the total expenses for each category and for the entire month. Generate a summary report of expenses (e.g., total, by category). Use a HashMap or ArrayList to store expenses and categories. 2. Implement methods to add, delete, and view expenses. Create reports that show the total expenses, categorized by month and type.	4
7	**Student Management System** Create a system where you can add, delete, and search students' records. The data will be stored in a text file and manipulated using Java's file handling features. Store student information (name, age, grade, etc.) in a .txt or .csv file. Perform CRUD (Create, Read, Update, Delete) operations on the file. Use a List to temporarily store the student data. Use collections such as ArrayList, HashMap, or TreeMap for sorting or searching student records.Add student details, Modify or delete student records, Search for students by name or grade, Save the data to a file and reload it at startup.	4
8	**Food Delivery System.** Create a Java program to simulate an online food delivery system where multiple delivery agents handle multiple orders simultaneously. Use multithreading to manage these tasks efficiently.	4
	Total Hours	**24**
Suggested Mini Projects List **Note:** **1. The Mini Project should be implemented in a group of 3-4 students.** **2. Students can select any one from the following mini project titles.** **3. Students can select a mini project title of their own choice.**		
P1	**Console Based Online Voting system.** Build a simple voting system where: Users can cast votes for a candidate (choose from a list of candidates). Track the number of votes for each candidate. Display the results after all votes are cast.Using a HashMap to store candidates and their vote counts., Implement methods to cast votes and display results,Validate that a voter can only vote once.	
P2	**Employee Management System**. Build an Employee Management System that can implement the following scenario. In a company, there are different types of employees (e.g., Manager, Engineer, and Intern). All employees have common properties like name, id, and salary, but Manager has additional properties like teamSize. Implement a class hierarchy with inheritance, where Manager and Engineer inherit from the Employee base class. Apply concept of Inheritance & method overriding	
P3	**To-do-List application.** Create a To-do-List application where users can manage their daily tasks effectively. The application can enable users to add, view, update, and delete tasks, helping them stay organized and productive. Users can enter tasks with specific details like task name, description, due date, and priority. The program could offer functionalities such as marking tasks as completed, displaying tasks based on their due dates or priorities, and allowing users to sort tasks in different ways (e.g., by deadline or importance). Use of file	

	handling, collections, and exception handling is mandatory.	6
P4	**Payment Gateway application.** Develop a Payment Gateway Application that can implement the following scenario. Design an abstract system for payment gateways. The payment gateways can be Stripe, PayPal, and Razorpay, and each gateway has a common functionality of initiatePayment(). However, each gateway will implement the payment initiation in a different way. Define an interface PaymentGateway and implement the details in each concrete class. Use Abstraction, Interface to implement the above scenario.	
P5	**E-commerce Order Management System.** Create an E-commerce Order Management System which includes a backend system to manage orders, track shipments, handle payments and report generation by using File handling and collections.	
P6	**Telephone directory.** Create a contact management system to Add, update, and delete contacts (name, phone number, email). Search for contacts by name or phone number. Display a list of all contacts. Use ArrayList or HashMap to store contacts. Optionally, store contacts in a text file for persistence.	
P7	**Banking System.** Implement the system that handles multiple simultaneous banking transactions using multithreading. A BankAccount class should maintain account details like accountNumber and balance. The balance must not become negative during transactions. Implement two types of transactions: withdraw and deposit. Multiple users (threads) can perform these operations simultaneously. Display the account balance after each transaction. Handle invalid transactions (e.g., withdraw amount greater than the balance) by printing an error message. Synchronize the critical section to avoid race conditions. **Example Execution:** Initial Balance: ₹10,000 User 1 tries to withdraw ₹6,000. User 2 tries to deposit ₹5,000 simultaneously. User 3 tries to withdraw ₹12,000 (invalid transaction).	
P8	**Movie Rental System** Build a movie rental system where users can search for movies, rent them, and track rental transactions. Data about movies and rental records will be stored in files. Store movie details (title, genre, director, release year) in a .txt or .csv file. Maintain a rental log (user, movie, rental date, return date) in a separate file. Use HashMap for storing movies and their details by title or ID. Use ArrayList to store the rental transaction history. Use HashSet to track movies that have been rented by users to avoid duplication. Perform following operations: ● View a list of available movies. ● Rent movies by users, saving transaction details. ● Search for movies by genre, title, or director	
	Total Hours	**30**

Guidelines for Laboratory Conduction:

1. All the assignments on all concepts are mandatory and should be implemented on coding platforms such as Leetcode, GitHub.
2. Operating System recommended: - 64-bit Open-source Linux or its derivative.
3. Programming tools recommended: - Eclipse/Geany, Apache

Guidelines for Students:

3. The laboratory assignments are to be submitted by students in the form of a journal.

4. Journal consists of Vision Mission of Institute & Department, certificate, Index table of contents and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept, algorithm/pseudocode, sample input and expected output, conclusion).

Guidelines for Laboratory Assessment:

1. Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of students.
2. Each Laboratory assignment assessment should assign grade/marks based on rubrics with appropriate weightage.
3. Suggested rubrics for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness.

Text Books

1. Herbert Schildt, "The Complete Reference Java", 13th Ed, TMH,ISBN: 978-0-07-180856-9.

Reference Books

1. Dr. R. Nageshwar Rao, "Core Java: An Integrated Approach",8th Ed, Dreamtech Press,ISBN 9789351199250

Online References

9. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01384205303214080010965_shared/overview
10. https://spoken-tutorial.org/tutorial-search/?search_foss=Java&search_language=
11. https://www.codechef.com/roadmap/java-dsa
12. NPTEL Course: Prof. Debasis Samanta, IIT Kharagpur, "Programming in Java" https://nptel.ac.in/courses/106105191

Coding Practice:

1. https://www.hackerrank.com/domains/java
2. https://leetcode.com/problemset

Second Year B.Tech Computer Engineering

Semester- IV

Course Code: CE24VSE257 **Course Name: Web Technology**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	4	-	-	-	-	-	50	-	-	50	-	2	-	2

Course Objectives:

The course aims :

- To gain a solid understanding of web essentials and markup languages.
- To apply client-side technologies effectively in web development.
- To utilize server-side technologies for building dynamic web applications.
- To comprehend web services and frameworks for efficient web development.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Understand the core concepts of web development, including HTML, CSS,Bootstrap and their role in creating dynamic web pages.

CO2: Apply JavaScript, and DOM to design responsive and interactive web pages that work across various devices.

CO3: Apply jQuery for DOM manipulation, events, and effects.

CO4: Analyze security risks in web applications, implementing best practices to secure applications from common vulnerabilities.

CO5: Apply server side technology PHP to create Dynamic web pages

Sr. No.	Name of the Experiment	Duration (Hrs.)
1	**Create a Webpage Using HTML**: 1. Design a webpage that includes structured content like headings, paragraphs, lists, tables, forms, and images. Ensure semantic use of HTML tags to maintain accessibility and SEO best practices. 2. Provide a description of the layout and structure of the page.	4
2	**Style the Webpage Using CSS**: 1. Apply various CSS properties to style the webpage. Include text formatting, background colors, borders, margins, and padding. 2. Implement the CSS Box Model to control element spacing and positioning.	4
3	**Enhance the Webpage Using JavaScript**: 1. Add interactivity to your webpage by writing JavaScript code that can manipulate the DOM, such as changing element styles, displaying messages, or hiding/showing content based on user actions. 2. Implement basic event handling (e.g., button clicks, form submissions).	4
4	**Implement an Web Application using Bootstrap:** To design a responsive web page using Bootstrap's grid system, components, and utilities. By the end of the lab, students will be familiar with key Bootstrap features and able to create a visually appealing and functional web layout.	4
5	**Integrate jQuery** 1. Use jQuery to simplify DOM manipulation and enhance the user experience with animations and effects. 2. Handle events like clicks and hover to change content dynamically.	4
6	**XML Data Handling:**	4

	1. Create an XML document to store structured data (e.g., list of products,users, or events) and process it using JavaScript or PHP. 2. Manipulate the XML document (e.g., read and modify elements) using DOM methods.	
7	**Content Management System (CMS)** **Set Up WordPress CMS**: 1. Explore the WordPress dashboard, create posts and pages, and install plugins. 2. Customize a WordPress theme by editing the CSS or creating a child theme.	4
8	**Create a PHP-based Web Application**: 1. Develop a PHP-based web application that accepts user input via an HTML form and processes it on the server side. For instance, create a contact form that sends the user's data to the server. 2. Integrate basic form validation to ensure secure data input.	4
	Total Hours	**32**
Suggested Mini Projects List **Note:** **1. The Mini Project should be implemented in a group of 3-4 students.** **2. Students can select any one from the following mini project titles.** **3. Students can select a mini project title of their own choice.**		
P1	**Personal Portfolio Website** Description: Create a personal portfolio website that showcases a student's skills, projects, and resume. It should include features like a contact form, an interactive design, and sections for skills, projects, and blogs.	28
P2	**Online Resume Builder** Description: Develop a web application that allows users to create, customize, and download their resumes in different formats (e.g., PDF, Word). It should offer pre-designed templates and allow users to input personal details, skills, and work experience.	
P3	**Task Management Web App** Description: Build a simple task management app where users can create, edit, delete, and mark tasks as completed. Add features like categorizing tasks, setting deadlines, and reminders.	
P4	**Online Book Store** Description: Create a simple online bookstore that allows users to browse books, view details, and add them to their cart. Include features like search, category filtering, and checkout (without actual payment gateway).	
P5	**Weather Dashboard** Description: Design a weather dashboard that allows users to check the weather of their location or any city by entering a city name. The app should display weather data such as temperature, humidity, wind speed, and weather conditions.	
P6	**Recipe Sharing Platform** Description: Develop a platform where users can share and search for recipes. It should allow users to post recipes with images, ingredients, and preparation steps, as well as rate and comment on others' recipes.	
P7	**Online Quiz Application** Description: Build a web-based quiz application where users can take quizzes on various topics. Features should include multiple choice questions, time limits, score tracking, and question randomization.	

P8	**E-learning Platform** Description: Create a basic e-learning platform that allows users to register, enroll in courses, and track their progress. Include features like quizzes, progress tracking, and course search.	
P9	**Movie Review Website** Description: Design a website where users can browse movies, read reviews, and submit their own reviews. Use an API to fetch movie data or build a small database for storing movie information.	
P10	**Budget Tracker** Description: Build a web application to track personal finances, where users can input income, expenses, and categorize their spending. Display charts and reports to show spending trends.	
	Total Hours	**60**

Guidelines for Laboratory Conduction:

1. The instructor is expected to design assignments by considering the prerequisites, technological aspects, utility, and current trends related to the topic. The assignment policy should cater to average students while also including elements that challenge and engage advanced learners. The use of open-source software is encouraged. Based on the concepts covered, students should work in groups of 2-3 to implement a mini project..
2. Operating System recommended: - 64-bit Open-source Linux or its derivative.
3. Programming tools recommended: - Eclipse/Geany, Apache/Tomcat server/PHP

Guidelines for Students:

1. The laboratory assignments are to be submitted by students in the form of a journal.
2. Journal consists of Vision Mission of Institute & Department, certificate, Index table of contents and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, Date of Completion, Assessment grade/marks and assessor's sign, Theory-Concept, algorithm/pseudocode, sample input and expected output, conclusion).

Guidelines for Laboratory Assessment:

1. Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of students.
2. Each Laboratory assignment assessment should assign grade/marks based on rubrics with appropriate weightage.
3. Suggested rubrics for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness.

Text Books

1. Robert W. Sebesta, "Programming the World Wide Web", 4th Edition, Pearson education,2008.
2. Web Technologies by Achyut Godbole and Atul Kahate, Publication: Tata McGraw-Hill Education Pvt. Ltd., ISBN 13: 9781259062681.
3. Web Technology: Theory and Practice by M. Srinivasan, Released June 2012, Publisher(s):Pearson India, ISBN: 9788131774199.
4. "JavaScript & JQuery: The Missing Manual" David Sawyer McFarland,O'Reilly Media Edition: 2nd Edition (2014)

Reference Books

1. Jon Duckett'HTML and CSS: Design and Build Websites',Wiley,2nd Edition,2022.
2. Douglas Crockford ,JavaScript: The Good Parts,,O'Reilly Media,1st Edition,2008.
3. Bryan Sullivan and Vincent Liu,Web Security,McGraw-Hill Education,1st Edition,2011.
4. Erik T. Ray,Learning XML,O'Reilly Media,4th Edition,2019.
5. Steven Holzner,PHP and MySQL: The Complete Reference,McGraw-Hill Education,1st Edition,2007
6. Web Technologies: A Computer Science Perspective, Jeffrey C.JacksonSecond Edition,

Pearson Education, 2007, ISBN 978-0131856035.

Online References

1. https://www.w3.org/html/
2. HTML, The Complete Reference http://www.htmlref.com/
3. http://w3schools.org/
4. http://php.net/
5. https://jquery.com/
6. https://developer.mozilla.org/en-US/docs/AJAX
7. http://www.tutorialspoint.com/css/

Second Year B.Tech Computer Engineering

Semester- IV

Course Code: CE24VEC258 **Course Name: OS Workshop**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	-	2	-	-	-	-	50	-	-	50	-	-	2	2

Course Objectives:

The Course aims :

- To understand the Basics of Linux Operating System.
- To demonstrate basic file and directory management commands.
- To understand roles and responsibilities of system administrators.

Course Outcomes:

After learning the course, the students will be able to:

CO1: Install, initiate and explore the Linux operating system.

CO2: Design a file system and directories to organize the data using different commands.

CO3: Develop shell scripts for automating system administration tasks.

Contents

1	**Introduction to Linux** Basic concepts : Operating system, services provided by operating system, the kernel and shells **Bootstrapping:** booting process, Bootloader: BIOS vs GRUB, startup scripts **Linux Filesystem:** Different directories and their usage: tree
2	**Managing Basic Storage** **Partitions:** View Partitions , Brief about EXT4 and XFS filesystem, Formatting Partitions , Mounting Partitions, Viewing the Mounts, Mounting Partitions at boot time **Managing Storage Stack:** Creating and Viewing the Logical Volumes, Extending and Reducing the Logical Volumes, Deleting Logical Volumes
3	**System Administration** Role of System administrator, Super user control, System time and date, Configuring directories and files, System logs, system auditing system, Linux Permissions: Read, Write, Execute permissions on file and directories. **Managing users:** user configuration files, password files, Managing users, Adding and removing users: Useradd , usermod, userdel, Managing groups: groupadd, groupmod, groupdel, **Access Control:** chmod, chgrp, setting permissions
4	**File Systems and Directories** **File systems:** Listing, displaying and printing files: ls,cat,more,less and **Managing Directories:** mkdir, rmdir, ls, cd and pwd, Creating and deleting directories, displaying directory contents, moving through directories , obtaining current working directory-getcwd **File and directory operations:** find, cp, mv, rm and ln, grep
5	**Shell Programming:** Introduction, Pipes and redirection, shell as scripting language, Shell syntax: variables, conditions, control structures, functions, commands, arithmetic in shell, command execution, Here documents, shell script examples. **Secure Shell:** Enabling the secure shell service(SSHD), Starting/Stopping the service, Allowing/Denying root user for SSH access, Passwordless SSH configuration.

List of Tutorial (Any 8)

Sr. No.	Name of the Tutorial	Duration (Hrs.)
1	Install Linux operating system in dual boot or standalone environment. Source: https://ubuntu.com/download/desktop (64-bit PC (AMD64) desktop image)	2
2	Install Vim editor to execute basic commands. Write a script to print Hello World using echo command display today's date and user.	2
3	List all files in /usr/bin and /usr/sbin that start with non-letter characters with one ls command	2
4	Create a text file describing your hobby. Save it with the .txt extension. Create a script that checks and removes duplicate lines in a text file. Also search for the keyword "hobby" in the file and replace it with "interest".	2
5	Find out the IP address of a given desktop machine (ifconfig). Check whether the machine is connected to the campus network and internet (ping Gateway and DNS)	2
6	Suppose you have users called user1, user2, and user3, and the groups demo1 and demo2. The demo1 group contains user1 and user2, and demo2 contains user2 and user3. You need to set these permissions for a shared directory: user1: read and write user2: read user3: read and write and execute demo1: read and write demo2: read and write and execute other: read	2
7	Write a shell script to calculate different ways to arrange the letters in the word "candy". (Use factorials).	2
8	Write a shell script that generates a secure random password. It should be at least 10 characters long It should be a combination of letters and numbers. Should have capital characters Should have at least 2 special characters	2
9	Write a shell script that calculates the total size of all files in a directory.	2
10	Free Certification: https://www.redhat.com/en/services/training/getting-started-with-linux-fundamentals	8
	Total Hours	**26**

Text Books

1. Richard Petersen, "The complete reference Linux", Tata Mcgraw-hill edition, ISBN: 9780071492478.
2. Matthew stones, "Beginning linux programming", Wiley India Pvt. Limited, 2008 ; ISBN, 8126515716, 9788126515714.

Reference Books

1. Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley, Unix and Linux system administration handbook, 4 th edition, ISBN 9789353062859

Online References

1. NPTEL Course: Prof. Anand Iyer, IIT Madras, "Linux Programming & Scripting" (Link)

Second Year B.Tech Computer Engineering

Semester-IV

Course Code: SH24AEC202 **Course Name: Professional Communication and Personality Development**

Teaching Scheme (Hours/Week)					Examination Scheme						Credits			
L	P	T	OL	ODL	CIE	ETE	TW	OR	PR	TOTAL	L	P	T	TOTAL
-	-	2	-	-	-	-	50	-	-	50	-	-	2	2

Prerequisite: Basic understanding of LSRW skills, grammar, and vocabulary of the English Language.

Course Objectives:

- To develop students' rhetorical and persuasive abilities for effective oral communication and audience engagement.
- To familiarize students with key business and managerial terminologies essential for effective communication and decision-making.
- To strengthen analytical and critical thinking skills for constructing logical arguments and making informed decisions.
- To cultivate professional and technical writing competence, ensuring clarity, accuracy, and etiquette in communication.

Course Outcomes:

After learning the course, the students will be able to:

CO1 Demonstrate effective rhetoric, persuasive skills, audience engagement, and adaptability in communication.

CO2: Apply visual aids, body language, voice modulation, and confidently manage audience questions and maintain engagement throughout.

CO3: Analyze information and construct logical arguments to solve problems and make informed Decisions.

CO4: Exhibit professional and technical writing skills and avoid common pitfalls, ensuring clear, polite, and informative communication.

Contents

1	**Public Speaking** Using Persuasive language and understanding the context in public speaking Controlling nervousness, building confidence, stage presence, poise, and impact Employing voice modulation, pace, volume & pitch.
2	**Presentation Skills** Structuring a presentation & using visual aids Understanding audience connection strategies & feedback Handling Q&A sessions with confidence.
3	**Critical Thinking** Working on Case Studies and Situational Analysis Structuring the argument and handling rebuttal Analyzing logical fallacies and cognitive biases.
4	**Technical Writing** Structuring emails for different purposes (requests, complaints, updates, and follow-ups) Avoiding common mistakes and making the right use of CC, BCC, and reply-all options Blog/article/ newsletter/survey report format and examples.

5	**Group Discussion** Implementing coherence and cohesion in structuring the argument Integrating non-verbal communication in the conversation Managing disagreement and aggression in the discussion.

List of Tutorials

Sr. No.	Name of the Tutorial	Duration (Hrs.)
1	Ice-breaking Session	2
2	Storytelling	4
3	Idea Presentation	4
4	Advertisement Creation	4
5	Case Study Analysis	2
6	Debate	2
7	Email Writing (Professional & Personal)	2
8	Blog Writing	2
9	Survey Reports	2
10	Group Discussion	2
	Total Hours	**26**

Text Books

1. Stephen E. Lucas, *The Art of Public Speaking*, McGraw Hill Publication, ISBN-10: 1260914275 | ISBN-13: 9781260914276.
2. Jaishri Jethwaney and Sanjay Bhargava, *Corporate Communication: Principles and Practices*.
3. Dr. Ranjit Singh, *Critical Thinking and Problem Solving*.
4. Rajendra Pal and J. S. Korlahalli, *Essentials of Business Communication*, Sultan Chand & Sons, 2011. ISBN: 8180547299, 9788180547294.

Reference Books

1. Shiv Khera, *You Can Win*, Bloomsbury India, ISBN-10: 9382951717 · ISBN-13: 9789832951711.
2. Carmine Gallo, *Talk Like TED*, Macmillan Business.
3. Garr Reynolds, *Presentation Zen: Simple Ideas on Presentation Design and Delivery*, New Riders.
4. Peter Facione and Carol Ann Gittens, *Think Critically*.
5. Lewis Vaughn, *The Power of Critical Thinking*, Oxford University Press.
6. Emily Post and Peter Post, *The Etiquette Advantage in Business: Personal Skills for Professional Success*.

Online References

1. NPTEL Course: *Technical English for Engineers* – https://onlinecourses.nptel.ac.in/noc20_hs56/preview
2. SWAYAM Course: *English for Research Paper Writing* – https://onlinecourses.swayam2.ac.in/ntr24_ed15/previe
3. SWAYAM Course: *Personality Development and Communication Skills* – https://onlinecourses.swayam2.ac.in/cec22_cm03/preview