Table III: Credit structure -Semester-III

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
CEC301	Engineering Mathematics III	03	-	01	03	-	01	04
CEC302	Basic Electronics and Logic Design	03		-	03		-	03
CEC303	Data Structures	03		-	03		-	03
CEC304	Computer Organization and Architecture	03		-	03		-	03
CEC305	Database Management Systems	03		-	03		-	03
CEL301	Basic Electronics and Logic Design Lab		02			01		01
CEL302	Data Structures Lab		02			01		01
CEL303	Computer Organization and Architecture Lab		02			01		01
CEL304	Database Management Systems Lab		02			01		01
CESL301	Skill Based Lab 1 - Object Oriented Programming Methodology	-	04	-	-	02	-	02
CEMP301	Mini Project I		02			01		01
*HBSO301	Sustainable Business Strategy Program(online)	-	-	-	01	-	-	01*
AUC301	Constitution of India	01	-	-	-	-	-	-
	Total	**16**	**14**	**01**	**16**	**07**	**01**	**23**

Table IV: Credit structure -Semester-IV

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	PR	Tutorial	Total
CEC401	Discrete Mathematics	03	-	01	03	-	01	04
CEC402	Theoretical Computer Science	03		01	03		01	04
CEC403	Operating Systems	03		-	03		-	03
CEC404	Computer Networks			-	03		-	03
CEC405	Design and Analysis of Algorithm	03		-	03		-	03
CEL401	Operating Systems Lab		02			01		01
CEL402	Computer Network Lab		02			01		01
CEL403	Design and Analysis of Algorithm Lab		02			01		01
CESL401	Skill Based Lab II - Web and Mobile Technologies	-	04	-	-	02	-	02
CEMP401	Mini Project II		04			02		02
*HBSO401	Entrepreneurship Essential program (online)	-	-	-	01	-	-	01*
AUC401	Sports and Yoga	01	-	-	-	-	-	-
	Total	**16**	**14**	**02**	**16**	**07**	**02**	**24**

Table III: Evaluation Scheme: Semester-III

Course Code	Course Name	IA1	IA2	AVG	MSE	ESE	Exam Hrs	TW	Pr	Or	Total
CEC301	Engineering Mathematics III	20	20	20	20	60	02				100
CEC302	Basic Electronics and Logic Design	20	20	20	20	60	02				100
CEC303	Data Structures	20	20	20	20	60	02				100
CEC304	Computer Organization and Architecture	20	20	20	20	60	02				100
CEC305	Database Management Systems	20	20	20	20	60	02				100
CEL301	Basic Electronics and Logic Design Lab							25		25	50
CEL302	Data Structures Lab							25	25		50
CEL303	Computer Organization and Architecture Lab							25		25	50
CEL304	Database Management Systems Lab							25	25		50
CESL301	Skill Based Lab 1 - Object Oriented Programming Methodology	-	-	-	-	-	-	25	25	-	50
	Total										**800**

Table IV: Evaluation Scheme: Semester-IV

Course Code	Course Name	IA1	IA2	AVG	MSE	ESE	Exam Hrs	TW	PR	OR	Total
CEC401	Discrete Mathematics	20	20	20	20	60	02				100
CEC402	Theoretical Computer Science	20	20	20	20	60	02				100
CEC403	Operating Systems	20	20	20	20	60	02				100
CEC404	Computer Networks	20	20	20	20	60	02				100
CEC405	Design and Analysis of Algorithm	20	20	20	20	60	02				100
CEL401	Operating Systems Lab							25		25	50
CEL402	Computer Network Lab							25	25		50
CEL403	Design and Analysis of Algorithm Lab							25	25		50
CESL401	Skill Based Lab II - Web and Mobile Technologies							25	25		50
CEMP401	Mini Project II							25		25	50
	Total										**750**

Semester III

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC301	**Engineering Mathematics III**	03	-	01	03	-	01	04

Subject Code	**Subject Name**	**Examination Scheme**								
		Theory Marks					**Term Work**	**Practical & Oral**	**Oral**	**Total**
		In-Sem Evaluations				**End-Sem Exam**				
		IA1	**IA2**	**Avg. of 2 IAs**	**Mid Sem Exam**					
CEC301	**Engineering Mathematics III**	20	20	20	20	60	--	--	--	100

Course Objectives:

1. To provide the requisite and relevant background necessary to understand other important engineering mathematics courses offered for Engineers.
2. To introduce three important topics of applied mathematics, viz, Z- Transform and Fourier Series and transform.
3. Fostering understanding through real-world statistical applications and to identify the direction and strength of a linear correlation between two factors and each source of variation in an analysis of regression.

Course Outcomes: At the end of the course learner will able to

1. Perform Gaussian elimination and apply to compute an LU decomposition.
2. Demonstrate ability to manipulate matrices and compute Eigen values and Eigen vectors.
3. Apply the Z- transform for analyze of discrete-time signals and systems.
4. Write given function in terms of sine and cosine terms in Fourier series and also to get knowledge in Fourier transforms.
5. Express the features of discrete and continuous random variables and apply the concept of probability distribution to the engineering problems.

6. Determine whether the correlation is significant and to calculate the simple linear regression equation for a set of data.

Prerequisites: EM I and EM II

Module	Contents	Hours	CO
1	**Linear Algebra I:** Vectors and linear dependence and independence, Solution of Linear system of equations by Gauss Elimination method, by LU Decomposition,	5	CO1
2.	**Linear Algebra II**: Eigen Values and Eigen vectors and their properties, Cayley-Hamilton theorem, Similarity of matrices and its properties, Diagonalization of matrices, Functions of square matrix, derogatory and non-derogatory matrices.	8	CO2
3	**Z-Transform:** Introduction, definition, properties (without proof), region of convergence, Inverse Z-Transform, convolution.	6	CO3
4	**Fourier Series and Transform:** Definition, Fourier series in (C, C+2L), Fourier complex integral, Fourier Transform, Properties (without proof), Fourier cosine and sine transform, convolution, Parseval's identity, Inverse Fourier Transform.	8	CO4
5	**Random Variables:** Discrete & continuous random variables, Probability mass function or density function and cumulative density Function, Mean, Variance, Poisson and Normal Distribution.	6	CO5
6	**Correlation and Regression**: Definition, scatter diagram, Karl Pearson's coefficient of correlation, Spearman's Rank correlation,. Lines of Regression	6	CO6

Text books:

1. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publication.
2. H.K. Das, "Advanced Engineering Mathematics", S. Chand, 2008.
3. R.K. Jain, S.R.K. Iyengar "Advanced Engineering Mathematics", Narosa Publication, 4th addition.

Reference Books:

1. B. V. Ramana, "Higher Engineering Mathematics", Tata Mc-Graw Hill Publication
2. Wylie and Barret, "Advanced Engineering Mathematics", Tata Mc-Graw Hill 6th Edition
3. Erwin Kreysizg, "Advanced Engineering Mathematics", John Wiley & Sons, Inc
4. Howard Anton and Christ Rorres. "Elementary Linear Algebra Application Version", 6th edition. John Wiley & Sons, INC.

Evaluation Scheme:

1. **In-Semester Assessment:**
   - Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
   - Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2. **End-Semester Examination:**
   - Question paper will comprise of full syllabus.
   - In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC302	**Basic Electronics and Logic Design**	03	-	-	03	-	-	03

Subject Code	**Subject Name**	**Examination Scheme**								
		Theory Marks					**Term Work**	**Practical & Oral**	**Oral**	**Total**
		In-Sem Evaluations				**End-Sem Exam**				
		IA1	**IA2**	**Avg. of 2 IAs**	**Mid Sem Exam**					
CEC302	**Basic Electronics and Logic Design**	20	20	20	20	60	-	-	-	100

Course Objectives:

1. To understand basic principles of analog and digital electronics.
2. To understand number systems representation and application of binary number applications.
3. To learn about logic gates and Boolean algebra.
4. To design and implementation of combinational circuits.
5. To design and implementation of sequential circuits.
6. To learn various types of programmable devices.

Course Outcomes: At the end of the course learner will able to

1. Understand the concept of basic electronics.
2. Understand number representation and perform arithmetic operations.
3. Design minimized Boolean expression through different minimization techniques.
4. Design and construct combinational circuits.
5. Design and construct sequential circuits.
6. Understand various types of programmable devices.

Prerequisites: Basic Electrical Engineering, Basic Mathematics.

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Basic Electronics and Logic families	Fundamentals of analog electronics devices: Diode, BJT, Operational amplifier, Oscillators. Basics of TTL, CMOS, Interfacing of TTL to CMOS and vice versa.	4	CO1
2	Number Systems	Difference between Analog and Digital Circuit. Introduction to different number systems and base conversion's, representation of negative numbers, compliment arithmetic, classification of codes.	6	CO2
3	Logic Gates and Boolean Algebra	Introduction to digital logic gates, Boolean Algebra Theorems and Properties, Standard SOP and POS form, Reduction of Boolean functions using Algebraic method, K -map method, Quine Mc Cluskey.	7	CO3
4	Combinational Circuits	Introduction, Half/Full Adder, Half/Full Subtractor, Four Bit Binary Adder, One digit BCD Adder, code conversion, Encoder and Decoder, Multiplexers and Demultiplexers, Decoders, Binary Comparator, ALU IC74181.	9	CO4
5	Sequential Circuits	Introduction, Flip-flop-types and conversion, master slave flip flop, Register: Shift register and its types, Bi-directional and universal shift register. Counters: Design of various synchronous and asynchronous.	9	CO5
6	Programmable Logic Devices	Introduction to programmable Logic Devices, ROM, RAM, SRAM, PLA, CPLD and FPGA architecture.	4	CO6

Text Books:

1. R. P. Jain, "Modern Digital Electronics", Tata McGraw Hill.
2. M. Morris Mano, "Digital Logic and computer Design", PHI
3. Anil K. Maini, Digital Electronics, Principles, Devices and Applications, Wiley
4. V.K Mehta, Rohit Mehta, "Principles of Electrical Engineering and Electronics ",S Chand Publication

Reference Books:

1. John F. Wakerley, "Digital Design Principles and Practices", 3rd Edition Updated, Pearson Education, Singapore, 2002.
2. Holdsworth and R. C. Woods, "Digital Logic Design", 4thEdition, Newnes, 2002

Evaluation Scheme:

1 In-Semester Assessment:

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 End-Semester Examination:

- Question paper will comprise of full syllabus.
- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC303	**Data Structures**	03	-	-	03			03

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC303	**Data Structures**	20	20	20	20	60	--	--	--	100

Course Objectives:

1. To design and implement various data structures.
2. To introduce various techniques for representation of data in the real world.
3. To teach various searching and hashing techniques.
4. To solve problems using linear and nonlinear data structures.

Course Outcomes: At the end of the course learner will able to

1. Differentiate primitive and non-primitive structures.
2. Handle operations like insertion, deletion, searching and traversing on various linear data structures.
3. Handle operations like insertion, deletion, searching and traversing on various non-linear data structures.
4. Apply learned concepts in various domains like DBMS, Compiler Construction
5. Demonstrate knowledge of various sorting and searching techniques.
6. Apply appropriate data structure for solving real world problem.

Prerequisites:

Programming in C

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Introduction to Data Structures	Introduction, Classification of Data Structures, Operations on Data Structures, Concept of ADT, Arrays, Recursion	02	CO1
2	Stack and Queues	Stack: ADT, Representation, Operations, Applications of Stack. ADT of Queue, Representation of Queue, Operations on Queue, Circular Queue, Priority Queue, Double Ended Queue Applications of Queue.	05	CO2, CO4, CO6
3	Linked List	ADT of Linked List, Representation of Linked List, Linked List v/s Array, Circular Linked List, Doubly Linked List Applications of Linked List.	07	CO2, CO4, CO6
4	Trees	Introduction, Binary Tree, Representation of Binary Trees, Types of Binary Tree, Binary Tree Traversals, Binary Search Tree, Applications – Expression Tree, Huffman Encoding. B/B+ Tree, AVL Tree, Splay Tree, Tries, Red-Black Trees, 2-3 Trees	11	CO3, CO4, CO6
5	Graphs	Introduction, Representation of Graph, Graph Traversals – Depth First Search (DFS) and Breadth First Search (BFS), Applications – Topological Sorting Minimum Spanning Tree: Prim's and Kruskal's Algorithm	06	CO3, CO4, CO6
6	Sorting and Searching	Selection Sort, Insertion Sort, Quick Sort, Merge Sort. Linear Search, Binary Search, Hashing – Hash Tables, Hash Functions, Collision Resolution.	08	CO5

Text Books:

1. Data Structures using C, Reema Thareja, Oxford
2. Data Structures Using C, Aaron M Tenenbaum, Yedidyah Langsam, Moshe J Augenstein, Pearson

Reference Books:

1. Data Structures and Program Design by Kruse et. al, PHI
2. Data Structure Using C, Balagurusamy.

Evaluation Scheme:

1 In-Semester Assessment:

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 End-Semester Examination:

- Question paper will comprise of full syllabus.
- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC304	**Computer Organization and Architecture**	03	--	-	03	--	-	03

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC304	**Computer Organization and Architecture**	20	20	20	20	60	--	--	--	100

Course Objectives:

1. To have a detailed understanding of the basic structure and working of a computer and arithmetic operations.
2. To discuss in detail arithmetic operations in digital system.
3. To discuss generation of control signals and different ways of communication with I/O devices.
4. To analyze processor performance improvement using instruction level parallelism

Course Outcomes: At the end of the course learner will able to

1. Understand basic structure of computer and perform computer arithmetic operations.
2. Demonstrate the arithmetic algorithms.
3. Understand control unit operation.
4. Understand the concept of cache mapping techniques.
5. Understand the concepts of I/O organization and conceptualize instruction level parallelism.
6. Describe the concepts of parallel processing.

Prerequisites:

Fundamentals of Computers

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Introduction to computer architecture	Difference between CO & CA, Von-Neuman Model, System Bus, Data Types Instruction Cycle, Instruction cycle with interrupt,	04	CO1
2	Data Representation	Fixed Point Representation, Floating - Point Representation (IEEE-754), Addition and subtraction, Multiplication Algorithms (Booth Multiplication Algorithm), Multiplication Algorithms (Booth Multiplication Algorithm), Division Algorithms – Restoring Division, Division Algorithms – Non-Restoring Division, Floating Point Arithmetic operations.	08	CO2
3	Control Unit	Overview of a Processor, Detailed Design of processing Stages, Hard-Wired Control Unit, Microprogrammed Processor, Micro assembly Language, Micro control Unit.	08	CO3
4	Memory Organization	Overview of the Memory System, Caches: Cache mapping Techniques, Coherence and Consistency, Cache Coherence Protocols, Details of Internal and External Memory System.	09	CO4
5	I\O Organization	External Devices, I/O Modules, Programmed I/O Interrupt-Driven I/O, Direct Memory Access, I/O Channels and Processors, Bus Arbitration Techniques.	05	CO5
6	Multiprocessor Systems	Overview of Parallel Programming, Multithreading and Vector Processing, Graphics Processors and Interconnects.	05	CO6

Text Books:

1. William Stallings, “Computer Organization and Architecture: Designing for Performance”, Pearson Publication, 10th Edition, 2013.
2. Smruti Ranjan Sarangi, “Computer Architecture and Organization”, McGraw-Hill, 2017.

Reference Books:

1. Morris Mano. “Computer System Architecture” Pearson Publication, 3rd Edition, 2007
2. Andrew S. Tanenbaum “Structured Computer Organization”, Pearson, Sixth Edition
3. V. Carl. Hamacher “Computer Organization”, McGraw-Hill

Evaluation Scheme:

1 In-Semester Assessment:

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 End-Semester Examination:

- Question paper will comprise of full syllabus.
- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC305	**Database Management Systems**	3	-	-	3	-	-	03

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC305	**Database Management Systems**	20	20	20	20	60	-	-	-	100

Course Objectives:

1. Learn and practice data modelling using the entity-relationship and developing database designs.
2. Understand the use of Structured Query Language (SQL) and learn SQL syntax.
3. Apply normalization techniques to normalize the database.
4. Understand the needs of database processing and learn techniques for controlling the consequences of concurrent data access.
5. Learn advanced representations of databases suited for real-time applications.

Course Outcomes: At the end of the course learner will able to

1. Understand the fundamentals of database systems and design ER for the real life problem.
2. Convert conceptual model to relational model, formulate relational algebra queries.
3. Analyze and apply concept of normalization to relational database design.
4. Design and querying database using SQL.
5. Perform PL/SQL programming using concept of cursor management, Triggers, stored procedures and functions.
6. Understand the concept of transaction, concurrency and recovery management system.

Prerequisites:

Data Structures

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Introduction to Database	**Introduction to database systems:** Data models, architecture of a database, Challenges in building a DBMS. **Entity Relationship Model:** E/R diagram notation, **Extended Entity Relationship Model:** Specialization, Generalization and Aggregation.	06	CO1
2	Relational Model	**Relational Data Model:** Introduction to the Relational Model and concept of keys, Mapping the ER to the Relational Model. **Relational Algebra:** Relational Algebra Operations and Relational algebra queries.	07	CO2
3	Relational Database Design	**Normal Forms:** Importance of a good schema design, functional dependencies. 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.	06	CO3
4	Structured Query Language (SQL)	**Overview of SQL:** SQL statements (commands) – DDL, DML, DTL and DCL. Integrity Constrains, set and string operations, aggregate function, views, Joins, Nested and Complex queries.	08	CO4
5	Procedural Language/ SQL	**Introduction to PL/SQL, stored** Procedures, Functions and Database Triggers.	06	CO5
6	Transaction, Concurrency and Recovery System	**Introduction to Transactions** - Transaction States - ACID Properties, Concurrent Executions, Serializability- Conflict and View, Lock based Protocols and Recovery system in transactions	06	CO6

Text Books:

1. G. K. Gupta, “Database Management Systems”, McGraw – Hill.
2. Korth, Slberchatz,Sudarshan, “Database System Concepts”, 6th Edition, McGraw – Hill
3. Elmasri and Navathe, “Fundamentals of Database Systems”, 5th Edition, Pearson Education.

Reference Books:

1. Dr. P.S. Deshpande, “SQL and PL/SQL for Oracle” 10g, Black Book, Dreamtech Press.
2. Gillenson, Paulraj Ponniah, “Introduction to Database Management”, Wiley Publication.
3. Sharaman Shah, “Oracle for Professional”, SPD.
4. Raghu Ramkrishnan and Johannes Gehrke, “Database Management Systems”, TMH.

Evaluation Scheme:

1 In-Semester Assessment:

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 End-Semester Examination:

- Question paper will comprise of full syllabus.
- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEL301	**Basic Electronics and Logic Design Lab**	-	02	-	-	01	-	01

Lab Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End-Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEL301	**Basic Electronics and Logic Design Lab**	-	-	-	-	-	25	--	25	50

Course Objectives:

1. To introduce the fundamental concepts and methods for design of digital circuits and a pre-requisite for computer Organization and architecture, microprocessor systems.
2. To provide the concept of designing Combinational and sequential circuits.
3. To provide basic knowledge of programmable logic devices

Course Outcomes:

1. Study and analyze basics of TTL and CMOS Logic families
2. Understand and implement different number system and their conversions.
3. Understand the basics of various digital components and recognize its importance in computer architecture.
4. Understand the principles of design of combinational logic circuits using basic components.
5. Design and simulate the basic digital circuit.
6. Study the basic programmable logic devices

Experiment List

Experiment No.	Experiments Name	CO Mapping
1.	Study of basic electronic components.	CO1
2.	To understand use of gray code in position encoders for angle-measuring devices and implement binary to gray conversion and vice versa	CO2
3.	To analyze and synthesize a Boolean expression using universal logic gates	CO3
4.	Design and implementation of Half /Full Adder using logic gates / universal gates.	CO4
5.	Suppose there are many points to measure and there is one voltmeter then design a multiplexer will randomly access those points and measure it depending on logical data input	CO4
6.	To understand the storage of digital data in RAM used in computers with the help of implementing Flip-Flops (JK and D).	CO5
7.	To realize decade counter (using IC 7490) used to count either time or Electronic Pulses.	CO5
8.	Study of building of reconfigurable digital circuits using programmable logic devices.	CO6

Text Books:

1. R. P. Jain, “Modern Digital Electronics”, Tata McGraw Hill.
2. M. Morris Mano, “Digital Logic and computer Design”, PHI
3. Anil K. Maini, Digital Electronics, Principles, Devices and Applications, Wiley
4. V.K Mehta, Rohit Mehta, “Principles of Electrical Engineering and Electronics “,SChand Publication.

Reference Books:

1. John F. Wakerley, “Digital Design Principles and Practices”, 3rd Edition Updated, Pearson Education, Singapore, 2002.
2. Holdsworth and R. C. Woods, “Digital Logic Design”, 4thEdition, Newnes, 2002

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be oral evaluation based on the laboratory work and the corresponding theory syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEL302	**Data structures Lab**		02	--	--	01	--	01

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEL302	**Data structures Lab**	--	--	--	--	--	25	25	--	50

Course Objectives:

1. To implement linear and non-linear data structures.
2. To solve problem involving stacks, queues, linked list, graphs and trees
3. To implement various sorting and searching techniques
4. To use appropriate data structures for real world applications.

Course Outcomes: At the end of the course learner will able to

1. To differentiate primitive and non-primitive structures.
2. To implement operations like insertion, deletion, searching and traversing on various linear data structures.
3. To implement operations like insertion, deletion, searching and traversing on various non- linear data structures.
4. To apply learned concepts in various domains like DBMS, Compiler Construction etc.
5. To demonstrate knowledge of various sorting and searching techniques.
6. To apply appropriate data structure for solving real world problem.

Prerequisites:

Programming in C

Suggested Experiments:

Experiment No.	Experiments Name	CO Mapping
1	Implementation of recursion function to solve tower of Hanoi Puzzle	CO1
2	Implementation of Stack using arrays	CO2
3	Implementation of INFIX to POSTFIX transformation	CO4
4	Evaluation of Postfix Expression	CO4
5	Implementation of Queue using arrays	CO2
6	Implementation of Circular Queue	CO2
7	Implementation of Priority Queue	CO2
8	Implementation of Singly Linked List	CO2
9	Implementation of Doubly Linked List	CO2
10	Implementation of Binary Search Tree	CO3
11	Implementation of Breadth First Search/Depth First Search	CO3
12	Implementation of Minimum Spanning Tree Algorithm	CO3
13	Implementation of Selection Sort/Merge Sort/Quick Sort	CO5
14	Implementation of Hashing Techniques	CO5
15	Case Study-Identify best suited data structures for solving real world problem.	CO6

Text Books:

1. Data Structures using C, Reema Thareja, Oxford
1. Data Structures Using C, Aaron M Tenenbaum, Yedidyah Langsam, Moshe J Augenstein, Pearson

Reference Books:

1. Data Structures and Program Design by Kruse et. al., PHI
2. Data Structure Using C, Balagurusamy.

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be Practical examination along with oral evaluation based on the laboratory work and the corresponding theory syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEL303	**Computer Organization and Architecture Lab**	-	2	-	-	1	-	1

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEL303	**Computer Organization and Architecture Lab**	-	-	-	-	-	25	--	25	50

Course Objectives:

1. To implement the operation of the arithmetic unit including the algorithms & implementation of fixed-point and floating-point addition, subtraction, multiplication & division
2. To study the different ways of communicating with I/O devices and standard I/O interfaces
3. To design memory subsystem including cache memory
4. To have through understanding of various computer buses

Course Outcomes: At the end of the course learner will able to

1. Assemble personal computer.
2. Design the basic building blocks of a computer: arithmetic-logic unit, registers, central processing unit and implement various algorithms like Booths algorithm for arithmetic operations.
3. Describe control unit design methods and operations.
4. Design the basic building blocks of a computer: memory.
5. Describe various I/O buses with merits and demerits.
6. Describe organization of various multicore processors.

List of Experiments

Experiment No.	Experiments Name	CO Mapping
1	Simulation of Ripple Carry Adder	CO2
2	Simulation of Carry Look Ahead Adder	CO2
3	Implement Booth's Algorithm Restoring Division algorithm.	CO2
4	Simulate Direct mapped cache design.	CO5
5	Simulate ALU design.	CO1
6	Presentation on Parallel Processors	CO4
7	Presentation on multi-processor architecture	CO4
8	Case Study on types of buses in computer architecture.	CO6
9	Construct the control unit for controlling the data path to generate and output the numbers from 1 to 10.	CO3
10	Case Study on multi core processor a) Intel Core 2 Duo processor. b) Sun Niagara c) Intel Nehalem quad core processor d) IBM: Power 6 e) AMD Dual core f) AMD Quad core	CO4

Text Books:

1. William Stallings, “Computer Organization and Architecture: Designing for Performance”, Pearson Publication, 10th Edition.
2. Smruti Ranjan Sarangi, “Computer Architecture and Organization”, McGraw-Hill, 2017.

Reference Books:

1. Morris Mano. “Computer System Architecture” Pearson Publication, 3rd Edition, 2007
2. Andrew S. Tanenbaum “Structured Computer Organization”, Pearson, Sixth Edition
3. Computer Organization -V. Carl. Hamacher (McGraw-Hill)

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be oral evaluation based on the laboratory work and the corresponding theory syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEL304	**Database Management Systems Lab**	-	2	-	-	1	-	1

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEL304	**Database Management Systems Lab**	--	--	--	--	--	25	25	--	50

Course Objectives:

1. To explore design and develop of relational model
2. To present SQL and procedural interfaces to SQL comprehensively
3. To introduce the concepts of transactions and transaction processing

Course Outcomes: At the end of the course learner will able to

1. Formulate problem Statement for real world example by using the fundamental concept of database system.Assemble personal computer.
2. Design ER /EER diagram and convert to relational model
3. Understand and Apply relational algebra, SQL Statements (DDL, DML, DCL and TCL commands) and constraints on the relations.
4. Write simple and complex queries.
5. Use PL / SQL Constructs.
6. Demonstrate the concept of concurrent transactions execution and frontend-backend connectivity.

Suggested List of Experiments:

Experiment No.	Experiments Name	CO Mapping
1	Identify the case study and detail statement of problem.	CO1
2	Design an Entity-Relationship (ER) / Extended Entity-Relationship (EER) Model.	CO2
3	Perform experiment on mapping of ER/EER to Relational schema model.	CO2
4	Create and populate database using Data Definition Language (DDL) and DML Commands for you‘re the specified System.	CO3
5	Apply Integrity Constraints for the specified system.	CO3
6	Perform Simple queries, string manipulation operations.	CO4
7	Implement and execute Nested queries and Complex queries	CO4
8	Perform and Implement Join operations.	CO4
9	Implement Views and Triggers, Stored Procedures.	CO5
10	Study and understand Transaction and Concurrency control.	CO6
11	Demonstrate Database connectivity	CO6
12	Implementation and demonstration of Transaction and Concurrency control techniquesusing locks.	CO6

Text Books:

1. G. K. Gupta, “Database Management Systems”, McGraw – Hill.
2. Korth, Slberchatz,Sudarshan, “Database System Concepts”, 6th Edition, McGraw – Hill
3. Elmasri and Navathe, “Fundamentals of Database Systems”, 5th Edition, Pearson Education.

Reference Books:

1. Dr. P.S. Deshpande, “SQL and PL/SQL for Oracle” 10g, Black Book, Dreamtech Press.
2. Gillenson, Paulraj Ponniah, “Introduction to Database Management”, Wiley Publication.
3. Sharaman Shah, “Oracle for Professional”, SPD.
4. Raghu Ramkrishnan and Johannes Gehrke, “Database Management Systems”, TMH.

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be practical examination and oral evaluation based on the laboratory work and the corresponding theory course syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CESL301	**Skill Based Lab I- Object Oriented Programming Methodology**		02+02*	--	--	02	--	02

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CESL301	**Skill Based Lab I - Object Oriented Programming Methodology**	--	--	--	--	--	25	25	--	50

Course Objectives:

1. To understand the object-oriented programming basics and its features.
2. Able to use a programming language to resolve problems.
3. To understand and apply Object Oriented Programming (OOP) principles using Java.
4. To study various java programming concept like multithreading, exception handling, packages etc.
5. To explain components of GUI based programming.

Course Outcomes:

1. Understand basics of OOP and apply fundamental programming constructs.
2. Understand and illustrate the features of classes and objects.
3. Elaborate the concept of strings, arrays and vectors.
4. Develop a program that implements the concept of inheritance, interfaces and packages.
5. Implement the notion of exception handling and multithreading.
6. Develop GUI based application.

Prerequisites: C Programming

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	**Introduction to OOP**	Basic Concepts of Object-Oriented Programming, Need of OOP, Object Oriented Programming Paradigm, Benefits of OOP, Introduction to OOP languages: C++ and Java. Features of Java Language, JDK environment, Basic Constructs of Java Programming	5	CO1
2	**Classes and objects**	Introduction to Class and Object Fundamentals, Method Definition, Access Specifier, Class scope and accessing class members, Implementation of class and object concept using Java language. Recursion , Argument Passing Mechanism, Method Overloading	6	CO2
3	**Arrays and String class**	Defining Arrays, String and StringBuffer Class	3	CO3
4	**Inheritance and Packages**	Importance of Inheritance, Different Types of Inheritance, Super keyword, Method Overriding, Abstract Class and Method, Built in and User Defined Packages in Java	5	CO4
5	**Exception Handling and Multithreading**	Error vs Exception, Concept of Exception Handling, Lifecycle of multithreading, Creating Threads	4	CO5
6	**GUI Programming**	Introduction to Java Swing class , Container Class, Difference Between AWT And Swing, Java Swing Class Hierarchy, Layout Manager, Designing GUI and building applications using swing in Java.	3	CO6

Text books:

1. Herbert Schildt, “JAVA: The Complete Reference”, Ninth Edition, Oracle Press.
2. Sachin Malhotra and Saurabh Chaudhary, “Programming in Java”, Oxford University Press, 2010

Reference Books:

1. Ivor Horton, “Beginning JAVA”, Wiley India.
2. Dietaland Dietal, “Java: How to Program”, PHI.
3. “JAVA Programming”, Black Book, Dreamtech Press.
4. “Learn to Master Java programming”, Staredu Solutions

Suggested List of Programs

Experiment No.	Experiments Name	CO Mapping
1.	Program on various methods to accept input data from user: Command line argument and Scanner class	CO1
2.	Program to demonstrate control structures: if-else, switch, for and while loop.	CO1
3.	Program to demonstrate Class and Object.	CO2
4.	Program to demonstrate method overloading.	CO2
5.	Program to demonstrate Constructors and overloading.	CO2
6.	Program on passing and returning an object as an argument.	CO2
7.	Program on 1D and 2D array.	CO3
8.	Program on Vector.	CO3
9.	Program on String and String Buffer class.	CO3
10.	Program on single and multilevel inheritance (Use super keyword).	CO4
11.	Program on interface demonstrating concept of multiple inheritance.	CO4
12.	Program to demonstrate try, catch, throw, throws and finally.	CO5
13.	Program to demonstrate user defined exception.	CO5
14.	Program on multithreading.	CO5
15.	Program to demonstrate component and container class to design GUI.	CO6
16.	Program to build any application with event handling and swing class.	CO6

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be Practical examination and oral evaluation based on the Theory and laboratory work.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEMP301	**Mini Project I**	-	02	-	-	01	-	01

Subject Code	**Subject Name**	**Examination Scheme**								
		Theory Marks					**Term Work**	**Practical & Oral**	**Oral**	**Total**
		In-Sem Evaluations				**End Sem Exam**				
		IA1	**IA2**	**Avg.of 2 IAs**	**Mid Sem Exam**					
CEMP301	**Mini Project I**	-	-	-	-	-	25	-	25	50

Prerequisites:

Basic Programming Knowledge

Lab Description:

The students will undergo the experience of developing a project based on any of the programming language learned during course by applying all stages of project development life cycle. They are expected to learn the skills of teamwork, survey of the problem domain and understand the issues to be addressed. Implement any real life application which will meet societal requirements. A quality report exploring technical report writing skills will be generated as part of activity.

Objectives:

1. To acquaint with the process of identifying the needs and converting it into the problem.
2. To familiarize the process of solving the problem in a group.
3. To acquaint with the process of applying basic engineering fundamentals to attempt solutions to the problems.
4. To inculcate the process of self-learning and research.

Outcomes: After successful completion of this lab, the student will be able to,

1. Identify problems based on societal /research needs.
2. Acquire practical skills of a chosen area of technology.

3. Practically apply the software project development life cycle stages namely Identify, Analyze, Formulate and implement software projects.
4. As an individual or in a team, the student will learn the team building and leadership qualities.
5. Enhance the effective communication skills, written skills and project presentation skills.
6. Analyze the impact of solutions in societal and environmental context for sustainable development.

Guidelines:

1. Students shall form a group of 3 to 4 students.
2. Students should do survey and identify needs, which shall be converted into problem statement for mini project in consultation with faculty supervisor/head of department/internal committee of faculties.
3. Faculty supervisor will give inputs to students during mini project activity; however, focus shall be on self-learning.
4. Students in a group shall understand problem effectively, propose multiple solutions and select best possible solution in consultation with guide / supervisor.
5. Students have to develop an application based on programming language they have learned during course.
6. Students can develop project using C along with animation, Java with database connectivity, GUI using Java, Dynamic web page creation using HTML and database connectivity etc.

Term Work:

The review/ progress monitoring committee shall be constituted by head of departments of the institute. There will be 2 reviews to evaluate the progress of mini project.

In continuous assessment focus shall also be on each individual student, assessment based on individual's contribution in group activity, their understanding, contribution and response to questions.

The Term work Marks are based on the weekly performance of the students, review performances, quality of the report, Oral performance and regularity.

End Semester Examination:

End of the semester, a panel of examiners will take a presentation and demonstration on the Mini Project along with oral evaluation.

Semester IV

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC401	**Discrete Mathematics**	03	--	01	03	--	1	04

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC401	**Discrete Mathematics**	20	20	20	20	60	--	--	--	100

Course Objectives:

1. To learn basics of set theory and mathematical logic in solving real life problems.
2. To gain knowledge of permutation, combination and discrete probability.
3. To learn relations and functions and its applications.
4. To interpret graph theory, algebraic system, coding theory and its applications.

Course Outcomes: At the end of the course learner will able to

1. Relate logical notation to define and reason about fundamental mathematical concepts of Logic.
2. Apply concepts of relations and functions in solving real life applications.
3. Illustrate the knowledge and skills obtained from elementary combinatorial processes such as permutations and combinations to investigate and solve a variety of discrete mathematical problems
4. Exploit the concepts and algorithms of graph theory in solving real life problems.
5. Illustrate the knowledge obtained from algebraic system to explore its applicability in cryptography, scientific computation etc.
6. Explore the concepts of coding theory used in Encoding and Decoding applications.

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Logic	**Logic:** Introduction, Statements, Connectives, Representation, Equivalence, Normal Forms, Predicates and Quantifiers, Mathematical Induction, Inference Theory of Predicate Calculus	06	CO1
2	Relation and Function	**Relation:** Definition, Types, Properties, Representation, Partial Ordering Relation, Hasse Diagram, Lattice, Equivalence Relation and classes, Recurrence Relation **Function:** Definition, Types, Composition, Identity, Inverse, Generating function	8	CO2
3	Discrete Probability	Introduction to Discrete Probability, Conditional Probability, Bayes' Theorem, Pigeonhole Principle, Applications	06	CO3
4	Graph Theory	Basic Terminology, Representation, Types of graphs, Eulerian and Hamiltonian graphs, Planar graph, Isomorphism, Tree, Applications	07	CO4
5	Algebraic System	Algebraic Systems, Semi Groups, Monoid, Groups, Abelian Groups, subgroups, Cyclic groups, Normal subgroups, Homomorphism, Isomorphism and Automorphism.	07	CO5
6	Coding Theory	Codes and group codes, Code generation, error detection and correction, maximum likelihood decoding function	05	CO6

Text Books:

1. BernadKolman, Robert Busby, Sharon Cutler Ross, Nadeem-ur-Rehman, "Discrete Mathematical Structures", Pearson Education.
2. K.H.Rosen, Discrete Mathematics and applications, fifth edition 2003, TataMcGraw Hill publishing Company.
3. C.L.Liu, Elements of Discrete Mathematics, second edition 1985, McGraw-Hill BookCompany. Reprinted 2000.

Reference Books:

1. J. P. Tremblay & R. Manohar, "Discrete Mathematical Structures with Application to Computer Science", MGH Publication
2. Seymour Lipschutz , Marc Lars Lipson," Discrete Mathematics" Schaum"sOutline, McGraw Hill Education.

Evaluation Scheme:

1 <u>In-Semester Assessment:</u>

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 <u>End-Semester Examination:</u>

- Question paper will comprise of full syllabus.
- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC402	**Theoretical Computer Science**	03	--	01	03	--	01	04

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC402	**Theoretical Computer Science**	20	20	20	20	60	--	--	--	100

Course Objectives:

1. Develop understanding of the mathematical foundations of computation including automata theory; the theory of formal languages and grammars.
2. Understand the concepts such as Regular expression, deterministic and non-deterministic finite automata and push down automata.
3. Develop understanding of different types of Turing machines and applications.
4. Understand the concept of Un-decidability

Course Outcomes: At the end of the course learner will able to

1. Use basic concepts of formal languages, finite automata and also understand equivalence of NFA and DFA.
2. Design Finite Automata's for different Regular Expressions and Languages
3. Construct context free grammar for various languages and recognize the ambiguous grammar.
4. Design pushdown automata for recognizing different languages.
5. Develop understanding of the computations performed by Turing Machine and also design Turing Machine for various languages.
6. Understand that there are limitations on what computers can do, and learn examples of unsolvable problems.

Prerequisites: Mathematical Foundation

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Finite Automata	**Introduction**. **Finite Automata:** Definition, Types, Designs, Equivalence, Minimization. **FA with Output:** Definitions, Types, Designs, Inter-conversion. **Applications**	9	CO1
2	Regular Language and Regular Expression	Definition, Identities, Equivalence of R.E and FA, Pumping lemma, Properties, Applications.	7	CO2
3	Grammar	**Grammar:** Definition, Chomsky hierarchy, Derivation, Parse tree, Ambiguous Grammar **Context Free Grammar:** Definition, Simplifications, Normal Forms. Context Free Language (CFL), Pumping Lemma, Properties.	9	CO3
4	Pushdown Automata	**Push Down Automata (PDA):** Definition, Notation, Design, Types, Applications.	5	CO4
5	Turing Machines	**Turing machine:** Definition, Design, Types, Universal Turing Machine.	6	CO5
6	Recursively Enumerable Languages	Introduction, Un-decidability, Halting Problem, Post correspondence problem, Rice Theorem.	3	CO6

Text Books:

1. J. C. Martin, ―Introduction to Languages and the Theory of Computation, Tata McGraw Hill.
2. Michael Sipser, ―Theory of Computation, Cengage learning.
3. Vivek Kulkarni, ―Theory of Computation, Oxford University Press, India.

Reference Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffery D. Ullman, ―Introduction to Automata Theory, Languages and Computation, Pearson Education.
2. Kavi Mahesh, ―Theory of Computation: A Problem-Solving Approach, Wiley-Ind

Evaluation Scheme:

1 <u>In-Semester Assessment:</u>

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 End-Semester Examination:

- Question paper will comprise of full syllabus.
- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC403	**Operating Systems**	03	-	-	03	-	-	03

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC403	**Operating Systems**	20	20	20	20	60	-	-	-	100

Course Objectives:

1. To introduce basic concepts and functions of operating systems.
2. To understand the concept of process, thread and resource management.
3. To understand the concepts of process synchronization and deadlock
4. To understand various Memory Management and Virtual Memory concept
5. To understand various IO Management and File management

Course Outcomes: At the end of the course learner will able to

1. Understand role of Operating System in terms of system call, system program and functions of operating system.
2. Apply and analyze the concept of a process and threads for resource management.
3. Apply and analyze process synchronization.
4. Apply and analyze deadlock handling
5. Understand the memory management and virtual memory
6. Apply and analyze different techniques of file and I/O management.

Prerequisites: Programming Language, Data Structures

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Introduction to Operating System	Definitions, Evolution and Types, Operating system structures, System Calls	04	CO1
2	Process and CPU scheduling	Process Concept, Uniprocessor Scheduling-Types, Threads	08	CO2
3	Process Synchronization and Deadlocks	Process Synchronization: Inter-process Communication, Process Synchronization: Hardware and software approaches, Classical Problems	06	CO3
4	Deadlocks	Deadlocks: Principles of Deadlock Prevention, Avoidance, Detection and Recovery	06	CO4
5	Memory Management and Virtual Memory	Memory Management: Basic concept, Memory Partitioning, Memory allocation Techniques, Fragmentation, Paging, Segmentation Virtual Memory: Demand Paging, Page Replacement Strategies: FIFO, Optimal, LRU, Thrashing	10	CO5
6	I/O and File Management	File Management: Concept of File, File Organization and Access, File Directories, File Sharing I/O Management: I/O devices and controllers, Interrupt handlers, Disk scheduling algorithms.	05	CO6

Text Books:

1. Abraham Silberschatz, Peter B. Galvin and Greg Gagne, Operating System Concepts, Wiley Publication, 9th Edition, 2016.
2. Andrew S. Tanenbaum, Modern Operating System, PHI Publication, 4th Edition, 2015.

Reference Books:

1. D. M. Dhamdhere, Systems Programming and Operating Systems, McGraw-Hill, 2nd Edition, 1996.
2. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition, 2014.
3. Maurice J. Bach, “Design of UNIX Operating System”, PHI.

Evaluation Scheme:

1. **In-Semester Assessment:**
   - Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
   - Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2. **End-Semester Examination:**
   - Question paper will comprise of full syllabus.
   - In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC404	**Computer Networks**	03	-	-	03	-	-	03

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC404	**Computer Networks**	20	20	20	20	60	-	-	-	100

Course Objectives:

1. To study the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model
2. To explore the fundamental components of Application layer
3. To understand the issues and challenges of Transport layer
4. To gain core knowledge of network layer protocols and IP addressing
5. To acquire knowledge of Data Link layer services and protocols
6. To know and discover applications of various application layer protocols

Course Outcomes: At the end of the course learner will able to

1. Explore the fundamental concepts computer networking and compare ISO – OSI model with TCP/IP model.
2. Evaluate and apply applications layer protocols
3. Demonstrate the knowledge of Transport layer functions and protocols
4. Design the network using IP addressing and sub netting / super netting schemes and analyze various routing algorithms and protocols at network layer.
5. Analyze Data Link layer protocols and congestion control algorithms.
6. Analyzed transmission media & explore

Prerequisites: Basic Electronics and Logic Design

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Introduction	Introduction: Basic Concepts and Network models, classification of networks, Network entities & layers , Reference model OSI and TCP/IP, design issues for the layers	04	CO1
2	Application layer	Application layer paradigms, Application layer protocols, Principles of protocols, HTTP, DNS protocols.	07	CO2
3	Transport layer	Transport layer Services & principles, multiplexing & demultiplexing applications, UDP, principles of reliable data transfer, TCP details, principles of congestion control, TCP congestion control.	08	CO3
4	Network layer	Network service model, routing principles, hierarchical routing, Internet Protocol (IP) & ICMP details, routing in the Internet, router internals, IPV6.	08	CO4
5	Data Link layer	Introduction, services, multiple access protocol, LAN addresses & Address Resolution Protocol , CSMA / CD protocol and its performance, PPP details.	08	CO5
6	Physical Layer	Introduction to Communication System, digital Communication, Electromagnetic Spectrum, Transmission Media, Wireless Transmission, Circuit and Packet Switching	04	CO6

Text Books:

1. James F. Kurose & K W Ross: Computer Networking: A Top Down Approach, Pearson Education (LPE)

Reference Books:

1. B.A. Forouzan, Data Communications and Networking, TMH (5e)
2. A.S. Tanenbaum, Computer Networks, Pearson Education, (4e)
3. Douglas E. Comer: Computer Network & Internet, Addison Wesley. 2. Andrew S. Tanenbaum : Computer Networks, PHI (5E)

Evaluation Scheme:

1 <u>In-Semester Assessment:</u>

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 <u>End-Semester Examination:</u>

- Question paper will comprise of full syllabus.

- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEC405	**Design and Analysis of Algorithm**	3		-	3	-	-	3

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEC405	**Design and Analysis of Algorithm**	20	20	20	20	60	-	-	-	100

Course Objectives:

1. To provide mathematical approach for Analysis of Algorithms
2. To solve problems using various strategies
3. To analyse strategies for solving problems not solvable in polynomial time.

Course Outcomes: At the end of the course learner will able to

1. Analyze the running time and space complexity of algorithms.
2. Describe, apply and analyze the complexity of divide and conquer strategy.
3. Describe, apply and analyze the complexity of greedy strategy.
4. Describe, apply and analyze the complexity of dynamic programming strategy.
5. Explain and apply backtracking, branch and bound and string-matching techniques to deal with some hard problems.
6. Describe the classes P, NP, and NP-Complete and be able to prove that a certain problem is NP-Complete.

Prerequisites:

Data structure

Sr. No.	Module	Detailed Content	Hours	CO Mapping
1	Introduction	Algorithm analysis, Asymptotic Notations, Recurrences. Methods for finding complexity of recursive algorithms, Analysis of selection and insertion sort.	07	CO1
2	Divide and Conquer	Binary search, Quick sort, Merge Sort Analysis of all algorithms.	04	CO2
	Greedy Design Strategies	Elements of Greedy Strategy, Minimum cost spanning tree algorithms, Dijkstra's Shortest Path Algorithm, Job Sequencing with deadline, Knapsack Problem	06	CO3
3	Dynamic Programming Approach	Elements of Dynamic programming, Multistage graphs, Bellman ford single source shortest path, Floyds Warshall's all pair shortest path algorithm, Travelling salesman problem, 0-1Knapsack Problem, LCS.	07	CO4
4	Backtracking and Branch-and-bound	N-queens problem, Sum of subsets, Graph coloring, Traveling Salesperson Problem, 15 Puzzle Problem.	06	CO5
5	String Matching Algorithms	Naïve, Rabin Karp, Finite automata, Knuth-Morris-Pratt.	04	CO5
6	NP-Completeness	P, NP, and NP-complete Problems. Vertex Cover, Hamiltonian Cycle and Traveling Salesman Problems.	05	CO6

Text Books:

1. T.H.Coreman , C.E. Leiserson,R.L. Rivest, and C. Stein, "Introduction to algorithms", 2nd edition , PHI publication 2005.
2. Ellis Horowitz, Sartaj Sahni , S. Rajsekaran. "Fundamentals of computer algorithms", University Press.

Reference Books:

1. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw- Hill Edition.
2. S. K. Basu, "Design Methods and Analysis of Algorithm", PHI.
3. John Kleinberg, Eva Tardos, "Algorithm Design", Pearson.
4. Michael T. Goodrich, Roberto Tamassia, "Algorithm Design", Wiley Publication

Evaluation Scheme:

1 <u>In-Semester Assessment:</u>

- Assessment consists of two Internal Assessments (IA1, IA2) out of which; one should be compulsory class test (on minimum 02 Modules) and the other is a class test / assignment on case studies / course project.
- Mid Semester Examination (MSE) will be based on 40-50% of the syllabus.

2 End-Semester Examination:

- Question paper will comprise of full syllabus.
- In the question paper, weightage of marks will be proportional to the total number of lecture hours as mentioned in the syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEL401	**Operating Systems Lab**	-	02	-	-	01	-	01

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEL401	**Operating Systems Lab**	-	-	-	-	-	25	-	25	50

Course Objectives:

1. To gain practical experience with designing and implementing concepts of operating systems such as system calls, CPU scheduling, process management, memory management, file systems and deadlock handling using C language in Linux environment.
2. To familiarize students with the architecture of Linux OS.
3. To provide necessary skills for developing and debugging programs in Linux environment.
4. To learn programmatically to implement simple operation system mechanisms.

Course Outcomes: At the end of the course, the students will be able to

1 Demonstrate basic Operating system Commands, Shell scripts, System Calls and API wrt Linux
2 Implement various process scheduling algorithms and evaluate their performance.
3 Implement and analyze concepts of synchronization and deadlocks.
4 Implement various Memory Management techniques and evaluate their performance.
5 Implement and analyze concepts of virtual memory.
6 Demonstrate and analyze concepts of file management and I/O management techniques.

Prerequisite:

Knowledge of Operating system principles

Suggested List of Experiments

Experiment No.	Experiments Name	CO Mapping
1.	**Explore Linux Commands** Explore usage of basic Linux Commands and system calls for file, directory and process management.	**CO1**
2.	**Linux shell script** a. Display OS version, release number, kernel version b. Display top 10 processes in descending order c. Display processes with highest memory usage. d. Display current logged in user and log name.	**CO1**
3.	**Linux- API** Implement any one basic commands of linux like ls, cp, mv and others using kernel APIs	**CO1**
4.	**Linux- Process** Create a child process in Linux using the fork system call	**CO2**
5.	**Process Management: Scheduling** a. Write a program to implement the concept of non-preemptive or preemptive scheduling algorithms b. Demonstrate the concept of non-preemptive or preemptive scheduling algorithms using CPU – OS Simulator	**CO2**
6.	**Process Management: Synchronization** Write a C program to implement Peterson's solution for critical section.	**CO3**
7.	**Process Management: Deadlock** Write a program to demonstrate the concept of deadlock avoidance through Banker's Algorithm	**CO3**
8.	**Memory Management**	**CO4**

	Write a program to demonstrate the concept of dynamic partitioning placement algorithms i.e. Best Fit, First Fit, Worst-Fit etc.	
9.	**Memory Management: Virtual Memory** Write a program to demonstrate the concept of page replacement policies for handling page faults	**CO5**
10.	**File Management & I/O Management** Write a program to demonstrate concept of disk scheduling.	**CO6**

Text Books:

3. Abraham Silberschatz, Peter B. Galvin and Greg Gagne, Operating System Concepts, Wiley Publication, 9th Edition, 2016.
4. Andrew S. Tanenbaum, Modern Operating System, PHI Publication, 4th Edition, 2015.

Reference Books:

4. D. M. Dhamdhere, Systems Programming and Operating Systems, McGraw-Hill, 2nd Edition, 1996.
5. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition, 2014.
6. Maurice J. Bach, “Design of UNIX Operating System”, PHI.

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be oral evaluation based on the laboratory work and the corresponding theory syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEL402	**Computer Network Lab**	-	02	-	-	01	-	01

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEL402	**Computer Network Lab**	-	-	-	-	-	25	25	-	50

Course Objectives:

1. To study the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model
2. To explore the fundamental components of Application layer
3. To understand the issues and challenges of Transport layer
4. To gain core knowledge of network layer protocols and IP addressing
5. To acquire knowledge of Data Link layer services and protocols
6. To know and discover applications of various application layer protocols

.

Course Outcomes: At the end of the course, the students will be able to

1. Demonstrate the concept of data communication with the help of networking commands and topology
2. Conceptualize data communication at application layer
3. Demonstrate the transport layer with the help of socket programming Techniques and protocols
4. Design the network using IP addressing and sub netting / super netting schemes using Network tools and simulators such as wireshark, NS2 etc.
5. Demonstrate the Data link layer with the help of error detection and correction
6. Explore the protocols at physical layer.

Experiment No.	Experiments Name	CO Mapping
1	Build a simple network topology and configure it for static routing protocol using packet tracer using Cisco Packet Tracer	CO1
2	Use basic networking commands (eg: ping, tracert, nslookup, netstat, ARP, RARP, ip, ifconfig, dig, route, etc) using Linux Command Prompt	CO1
3	Study on Bluetooth protocol stack	CO6
4	Implement the Hamming code using Java	CO5
5	Implement Stop and wait protocol / sliding window (selective repeat / Go back N). Using Netsim	CO5
6	Setup a network and configure IP addressing, subnetting, Masking using CISCO Packet Tracer.	CO4
7	Simulate congestion control (leaky bucket / token bucket) using C code/Netsim	CO4
8	Implement TCP or UDP Sockets using Java	CO3
9	Perform File Transfer and Access using FTP commands	CO2
10	Perform Remote login using Telnet server/ SSH server using Linux Command Prompt	CO2
11	1. Set up multiple IP addresses on a single LAN. 2. Using nestat and route commands of Linux, do the following: • View current routing table • Add and delete routes • Change default gateway 3. Perform packet filtering by enabling IP forwarding using IPtables in Linux.	CO4
12	Use Wireshark to understand the operation of TCP/IP layers : • Ethernet Layer : Frame header, Frame size etc.	CO3, CO4, CO5

	• Data Link Layer : MAC address, ARP (IP and MAC address binding) • Network Layer : IP Packet (header, fragmentation), ICMP (Query and Echo) • Transport Layer: TCP Ports, TCP handshake segments etc. • Application Layer: DHCP, FTP, HTTP header formats	
13	Case study on different network discovery using discovery tools (eg. Mrtg/Prtg)	CO4, CO5

Text Books:

1. James F. Kurose & K W Ross: Computer Networking: A Top Down Approach, Pearson Education (LPE)

Reference Books:

1. B.A. Forouzan, Data Communications and Networking, TMH (5e)
2. A.S. Tanenbaum, Computer Networks, Pearson Education, (4e)
3. Douglas E. Comer: Computer Network & Internet, Addison Wesley. 2. Andrew S. Tanenbaum : Computer Networks, PHI (5E)

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be Practical examination along with oral evaluation based on the laboratory work and the corresponding theory course Syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEL403	**Design and Analysis of Algorithm Lab**	--	02	--	--	01	--	01

Subject Code	**Subject Name**	**Examination Scheme**								
		Theory Marks					**Term Work**	**Practical & Oral**	**Oral**	**Total**
		In-Sem Evaluations				**End Sem Exam**				
		IA1	**IA2**	**Avg. of 2 IAs**	**Mid Sem Exam**					
CEL403	**Design and Analysis of Algorithm Lab**	--	--	--	--	--	25	25	--	50

Course Objectives:

1. To introduce the methods of designing and analyzing algorithms
2. Design and implement efficient algorithms for a specified application
3. Strengthen the ability to identify and apply the suitable algorithm for the given real-world problem.
4. Analyze worst-case running time of algorithms and understand fundamental algorithmic problems.

Course Outcomes : At the end of the course, the students will be able to

1. Analyze the running time and space complexity of algorithms using different methods.
2. Implement and analyze the complexity of algorithms by applying divide and conquer approach.
3. Implement and analyze the complexity of algorithms by applying greedy strategy.
4. Implement and analyze the complexity of algorithms by applying dynamic programming strategy.
5. Implement and analyze the complexity of algorithms by applying backtracking approach.
6. Implement and analyze the complexity of algorithms by applying string-matching techniques.

Prerequisites :

C Programming, Data Structures

Suggested Experiment List :

Experiment No.	Experiments Name	CO Mapping
1.	Program to implement Selection sort.	CO1
2.	Program to implement Insertion sort.	CO1
3.	Program to implement Binary Search using Divide and Conquer approach.	CO2
4.	Program to implement Merge Sort using Divide and Conquer approach.	CO2
5.	Program to implement Quick Sort using Divide and Conquer approach.	CO2
6.	Program to find minimum cost spanning trees using Kruskal and Prim's algorithm.	CO3
7.	Program to find single source shortest path using Dijkstra algorithm and by applying Greedy approach.	CO3
8.	Program to find optimal solution of Fractional Knapsack Problem by applying Greedy approach.	CO3
9.	Program to find single source shortest path using Bellman Ford algorithm and by applying Dynamic Programming approach.	CO4
10.	Program to find all pair shortest path using Floyd Warshall algorithm and by applying Dynamic Programming approach.	CO4
11.	Program to find optimal solution of 0/1 Knapsack Problem by applying Dynamic Programming approach.	CO4
12.	Program to Longest Common Subsequence by applying Dynamic Programming approach.	CO4
13.	Program to solve N-queen problem by applying Backtracking approach.	CO5
14.	Program to find sum of subsets by applying Backtracking approach.	CO5
15.	Program to solve graph colouring problem by applying Backtracking approach.	CO5
16.	Program to implement Naïve string-matching algorithms.	CO6
17.	Program to implement Rabin Karp algorithm.	CO6
18.	Program to implement Knuth-Morris-Pratt algorithm.	CO6

Text Books:

1. T.H.Coreman , C.E. Leiserson,R.L. Rivest, and C. Stein, “Introduction to algorithms”, 2nd edition , PHI publication 2005.
2. Ellis Horowitz, Sartaj Sahni , S. Rajsekaran. “Fundamentals of computer algorithms”, University Press.

Reference Books:

1. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, “Algorithms”, Tata McGraw- Hill Edition.
2. S. K. Basu, “Design Methods and Analysis of Algorithm”, PHI.
3. John Kleinberg, Eva Tardos, “Algorithm Design”, Pearson.
4. Michael T. Goodrich, Roberto Tamassia, “Algorithm Design”, Wiley Publication

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be Practical examination along with oral evaluation based on the laboratory work and the corresponding theory syllabus.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CESL401	**Skill Based Lab II- Web and Mobile Technologies**	--	04	-	--	02	-	02

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CESL401	**Skill Based Lab II - Web and Mobile Technologies**	-	-	-	-	-	25	25	--	50

Course Objectives:

1. To introduce programming languages and techniques associated with the World Wide Web.
2. To use tools for creating interactive web pages and mobile applications.
3. To introduce the client-server architecture.

Course Outcomes: At the end of the course learner will able to

1. Develop static web pages and apply styling effects to web pages using various styling tools.
2. Implement dynamic web applications using client side technologies and server side technologies
3. Understand the basics of XML, DTD and XSL and develop web pages using XML/XSLT.
4. Implement dynamic mobile applications using client side technologies and server side technologies.
5. Analyze end user requirements and create application using appropriate framework for Web development and Mobile APP development.

6. Implement security in web using Secure Socket Layer

Prerequisites: OOPM, DBMS

Experiment No.	Experiments Name	CO Mapping
1.	Create Home/Index page using HTML5 and apply styling to web page using CSS3.	CO1
2.	Creating page Layout and Site designs using designing tools like Bootstrap, etc.	CO1
3.	Form Design and Client Side Validation using Javascript and HTML5.	CO2
4.	Create animation using any animation technologies like Flash, Jquery	CO2
5.	Design registration form using PHP.	CO2
6.	Develop interactive web page with database connectivity using PHP.	CO2
7.	Develop XML web page using DTD and XML schema.	CO3
8.	Design a Web application using any one Framework e.g. Wordpress, Laravel, Django, Ruby on Rails, React.	CO4
9.	Install and configure Android Studio. Develop and application that uses GUI components, Font and Colors.	CO4
10.	Develop an application that uses Layout Managers and event listeners.	CO4
11.	Develop any one application using Android Development Tool- a) Calculator b) Alarm Clock	CO5
12.	Develop interactive application with database connectivity in Android.	CO5
13.	Integration of Third party services in Web Application like SMS gateway, Payment gateway, Email.	CO5

14.	Hosting the website with Domain Registration Process.	CO6
15.	Integration of SSL Certificate in web application.	CO6

Term Work:

The Term work Marks are based on the weekly experimental performance of the students, Oral performance and regularity in the lab.

Students are expected to be prepared for the lab ahead of time by referring the manual and perform the experiment under the guidance and discussion. Next week the experiment write-up to be corrected along with oral examination.

End Semester Examination:

End of the semester, there will be Practical examination and oral evaluation based on the Theory and laboratory work.

		Theory Hrs	Practical Hrs	Tutorial Hrs	Theory Credit	Practical/Oral Credit	Tutorial Credits	Total Credits
CEMP401	**Mini Project II**	-	04	-	-	02	-	02

Subject Code	Subject Name	Examination Scheme								
		Theory Marks					Term Work	Practical & Oral	Oral	Total
		In-Sem Evaluations				End Sem Exam				
		IA1	IA2	Avg. of 2 IAs	Mid Sem Exam					
CEMP401	**Mini Project II**	-	-	-	-	-	25	-	25	50

Prerequisites:

Basic Programming Knowledge

Lab Description:

The students will undergo the experience of developing a project based on any of the programming language learned during course by applying all stages of project development life cycle. They are expected to learn the skills of teamwork, survey of the problem domain and understand the issues to be addressed. Implement any real-life application which will meet societal requirements. A quality report exploring technical report writing skills will be generated as part of activity.

Objectives:

1. To acquaint with the process of identifying the needs and converting it into the problem.
2. To familiarize the process of solving the problem in a group.
3. To acquaint with the process of applying basic engineering fundamentals to attempt solutions to the problems.
4. To inculcate the process of self-learning and research.

Outcomes: After successful completion of this lab, the student will be able to

1. Identify real life problems.
2. Acquire practical knowledge within the chosen area of technology for project development.
3. Identify, Analyze, Formulate and handle web development/ Mobile development projects with a systematic approach.
4. As an individual or in a team, the student will learn the team building and leadership qualities.
5. Enhance the effective communication skills, written skills and project presentation skills.
6. Prepare a solution in the form of product, prototype development model, fabrication of setups, laboratory experiment development, process modification and development, software development, creating social awareness, etc.

Guidelines:

1. A project to be developed using Web and Mobile Technologies.
2. Students should form a group of 3 to 4 students.
3. Students should do survey and identify needs, which shall be converted into problem statement for mini project in consultation with faculty supervisor/head of department/internal committee of faculties.
4. Students has to develop an interactive dynamic application based on web technologies and mobile technologies they have learned during course like Bootstrap, PHP, Frameworks-WordPress, Laravel Framework, Django, Ruby on Rails, React, Node.js, Android Studio, Flutter.

Term Work:

The review/ progress monitoring committee shall be constituted by head of departments of the institute. There will be 2 reviews to evaluate the progress of mini project.

In continuous assessment focus shall also be on each individual student, assessment based on individual's contribution in group activity, their understanding, contribution and response to questions.

The Term work Marks are based on the weekly performance of the students, review performances, quality of the report, Oral performance and regularity.

End Semester Examination:

End of the semester, a panel of examiners will take a presentation and demonstration on the Mini Project along with oral evaluation.