Course Type	Course Code	Course Name	Credits
PCC	ITPCC301	ENGINEERING MATHEMATICS- III	03+01*

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Exam (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20 +25*	30	50	1.5	2	125

*For Tutorial

Pre-requisite :

1. BSC101 Engineering Mathematics-I
2. BSC204 Engineering Mathematics-II

Program Outcomes addressed :

1. PO1: Engineering knowledge
2. PO2: Problem analysis

Course Objectives :

1. To provide the basic knowledge on the concepts of Mathematics in the field of Engineering.
2. To build a foundation to the methodology necessary for solving problems by applying the knowledge of Mathematics to the field of Engineering.

Module	Details	Hrs
00.	**Course Introduction** Engineering Mathematics III is often a foundational course designed to provide students with the mathematical tools and concepts essential for various engineering disciplines. Engineering Mathematics III has many applications in Information Technology such as 1) Application of Vector space in Deep Learning, dimension reduction and Neural Networks 2) Application of Linear Mapping in Deep learning and Machine Learning. 3) Application of in Number theory in Cryptography.	**01**
01.	**Vector Space** ***Learning Objective/s:*** *To analyse the definition of basis and apply it to determine the basis of a Vector Space*	**07-09**
	Contents:	
	Vector Spaces, Examples of vector spaces, Subspace, Linear combinations, Spanning Set, Linear Independence and Dependence. Linear spans, Row Space of a matrix , Basis and Dimensions	

	Self-Learning Topics: *Independence and Dependence of functions*	
	Learning Outcomes : *A learner will be able to* *LO 1.1: Apply the axioms of closure, addition and scalar multiplication and prove that the given set of vectors is a Vector Space (P.I.- 1.1.1)* *LO 1.2: Identify the conditions of closure and scalar multiplication and prove that the given space is a Subspace. (P.I.- 2.1.3)* *LO 1.3: Applying the fundamentals of row reductions for linear independency in order to determine the span, basis and dimension of a vector space. (P.I.- 1.3.1)* *LO 1.4: Evaluate the statements given in a problem and thereby identify whether linear dependence or independence of a set is to be considered.(P.I.- 2.1.1)*	
02.	**Linear Mappings** ***Learning Objective/s:*** *To apply concepts of kernel and image of linear maps and be able to compute and analyse them for given linear maps.*	**06-08**
	Contents: Mappings, Function, Linear mapping (Linear transformations), Kernel and Image of a linear mapping, Rank and Nullity, Singular and non-singular mapping, Isomorphism.	
	Self-Learning Topics: *Operations with linear mapping*	
	Learning Outcomes : *A learner will be able to* *LO 2.1: Apply the properties of vector addition and scalar multiplication to examine whether the mapping is a linear mapping. (P.I.- 1.1.1)* *LO 2.2: Apply laws of matrix transformations to obtain the Rank and Nullity of a given mapping. (P.I.-1.2.1)* *LO 2.3: Examine whether the given mapping is an Isomorphism using the results of non-singularity and linear mapping. (P.I.-2.1.3)* *LO 2.4: Identify whether rows and columns are to be formed for a given linear transformation based on whether Image or Kernel is to be found or to determine its non-singularity. (P.I.-2.2.3)*	
03.	**Linear Mappings and Matrices** ***Learning Objective/s:*** *To analyse and compute the change of basis matrix for linear map.*	**06-08**
	Contents: Introduction, Matrix Representations of a linear operator, Change of Basis, Similarity, Matrices and general linear mapping.	
	Self-Learning Topics: *Composition mapping*	

	Learning Outcomes : *A learner will be able to* *LO 3.1: Apply fundamentals of addition and scalar multiplication of mapping to represent it in its matrix form. (P.I.-1.3.1)* *LO3.2: Apply rules of addition, multiplication ad inverses of matrices to check the similarity of two mappings. (P.I.-1.1.1)* *LO 3.3: Identify the domain and codomain given and determine the change of Basis Matrix from one coordinate system to the alternate coordinate system. (P.I.-2.1.3)* *LO 3.4: Identify the existing method of finding inverse from the change of basis matrix using similarity of matrices. (P.I.-2.2.3)*	
04.	**Inner Product spaces, Orthogonality** ***Learning Objective/s:*** *To analyse and apply Gram-Schmidt Technique to determine an Orthonormal Basis.*	**06-08**
	Contents: Introduction, Inner product spaces, Examples of Inner product spaces, Cauchy-Schwartz Inequality, Orthogonality, orthogonal sets and Basis Gram-Schmidt orthogonalization process	
	Self-Learning Topics: *Complex Inner Products Spaces*	
	Learning Outcomes : *A learner will be able to* *LO 4.1: Identify whether the given vector space is an Inner Product Space using linearity, Symmetric and Positive Definite Property of an Inner Product Space. (P.I.-2.1.3)* *LO 4.2: Prove Cauchy's Schwartz inequality using definitions describing the space. (P.I.-1.1.1)* *LO 4.3: Identify whether the set of vectors form an orthogonal or orthonormal basis using the standard operations of Inner Product Space. (P.I. -2.2.3)* *LO 4.4: Apply laws of dot product to determine the orthogonal and orthonormal basis using Gram Schmidt Process. (P.I. -1.2.1)*	
05.	**Number Theory** ***Learning Objective/s:*** *To identify and apply the appropriate theorem of Number Theory to solve the simultaneous system of congruence.*	**06-08**
	Contents:	
	Modular Arithmetic, Divisibility and Euclid Algorithm, Primes and Sieve of Eratosthenes, Euler's and Fermat Little Theorem, Congruence, Computing Inverse in congruence, Legendre and Jacobi Symbol, Chinese Remainder Theorem	
	Self-Learning Topics: *Testing for Primes, Prime Number Theorem*	

	Learning Outcomes : *A learner will be able to* *LO 5.1: Apply Euler's theorem or Fermat little theorem to solve congruent modulo equation. (P.I.-1.1.1)* *LO 5.2: Identify whether modular inverse or Euclidean algorithm is to be applied to the given Linear Diophantine Equation. (P.I.- 2.2.3)* *LO 5.3: Evaluate the information presented in a given problem and thereby identify whether Euler's theorem, Fermat's Theorem or Chinese Remainder Theorem is to be used. (P.I.-2.1.1)* *LO 5.4: Apply fundamental techniques of finding remainders to solve simultaneous linear congruence's using Chinese Remainder theorem. (P.I.-1.3.1)*	
06.	**Numerical Methods** ***Learning Objective/s:*** *To analyse and apply the appropriate numerical method to solve transcendental equation and system of simultaneous equations.* **Contents:** Solution of Transcendental Equations: Newton Ralphson method, Regula –Falsi Method, Solution of system of linear algebraic equations Gauss Jacobi Iteration Method, Gauss Seidel Iteration Method. ***Self-Learning Topics:*** *Bisection Method, Gauss Elimination Method.* ***Learning Outcomes :*** *A learner will be able to* *LO 6.1: Apply Newton Raphson method or Regula Falsi method to solve the transcendental equation or algebraic equations. (P.I.-1.1.1)* *LO 6.2: Apply the fundamental arithmetic techniques to solve the system of equations using Gauss Siedel or Gauss Jacobi methods. (P.I-1.3.1)* *LO 6.3: Identify the appropriate numerical method which can be applied to solve the given equation or system of equations. (P.I- 2.1.3)* *LO 6.4: Compare the results obtained by Gauss Siedel and Gauss Jordon methods and thereby make conclusions about its rate of convergence. (P.I.-2.2.4)*	**06-08**
	Course Conclusion Engineering Mathematics provides the language and framework through which engineers model, analyse, and optimize systems.	**01**
Total		**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.1.1	Apply the knowledge of discrete structures, linear algebra, statistics and numerical techniques to solve problems
1.2.1	Apply laws of natural science to an engineering problem
1.3.1	Apply engineering fundamentals

2.1.1 Evaluate problem statements and identifies objectives

2.1.3 Identify mathematical algorithmic knowledge that applies to a given problem

2.2.3 Identify existing solution/methods to solve the problem, including forming justified approximations and assumptions

2.2.4 Compare and contrast alternative solution/methods to select the best methods

Course Outcomes :

1. Apply the fundamentals of Vector Space to analyse the Basis of the Vector Space. *(LO 1.1, LO 1.2, LO 1.3, LO 1.4)*
2. Apply the fundamentals of Linear Mapping to identify the Kernel and Image of the Linear Mapping. *(LO 2.1, LO 2.2, LO 2.3, LO 2.4, LO 3.1, LO 3.2, LO 3.3, LO 3.4)*
3. Analyse the Inner Product Space and apply the properties to determine the orthogonal basis. *(LO 4.1, LO 4.2, LO 4.3, LO 4.4)*
4. Identify and apply the appropriate theorem of Number Theory to solve the congruent equation and the simultaneous congruent equations. *(LO 5.1, LO 5.2, LO 5.3, LO 5.4)*
5. Identify and apply appropriate numerical methods to solve System of equations. *(LO 6.1, LO 6.2, LO 6.3, LO 6.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC301.1	3	3	-	-	-	-	-	-	-	-	-
ITPCC301.2	3	3	-	-	-	-	-	-	-	-	-
ITPCC301.3	3	3	-	-	-	-	-	-	-	-	-
ITPCC301.4	3	3	-	-	-	-	-	-	-	-	-
ITPCC301.5	3	3	-	-	-	-	-	-	-	-	-
Average	**3**	**3**	-	-	-	-	-	-	-	-	-

Text Books :

1. C. L. Liu: Elements of Discrete Mathematics, 2nd edition, TMH
2. Cryptograph and Network Security by B. A. Forouzan & D. Mukhopadhyay, 11th edition, McGraw Hill Publication

Reference Books :

1. Elementary Number Theory and its applications by Kenneth H. Rosen, 5th edition, Addison Wesley Publication
2. Advanced Engg. Mathematics by C. Ray Wylie & Louis Barrett. TMH International Edition
3. Seymour Lipschutz “Beginning Linear Algebra” Schaum’s outline series, Mc-Graw Hill Publication

Other Resources :

1. NPTEL Course: Applied Linear Algebra IIT Madras: Dr Andrew Thangaraj: :-Web link- https://nptel.ac.in/courses/108106171
2. NPTEL Course: Computational Number Theory and Cryptography IIT Guwahati: Dr Pinaki Mishra:-Web link- https://nptel.ac.in/courses/106103015
3. NPTEL Course:NOC: Numerical Methods, IIT Roorkee Prof. Ameeya Kumar Nayak, Prof. Sanjeev Kumar - Web link- https://nptel.ac.in/courses/111107105

A. IN-SEMESTER ASSESSMENT (75 MARKS)

1. Continuous Assessment - Theory (20 MARKS)

One MCQ test as per Gate exam pattern/ level: 5 Marks
One Class test: 5 Marks
One Team-pair- Solo: 5 Marks
Regularity and attentiveness: 5 Marks

Continuous Assessment - Tutorial (25 MARKS)

Minimum six Tutorials: 20 Marks

Students must be encouraged to write at least 6 class tutorials. At least 6 Class tests will be conducted based on class tutorials on entire syllabus. Each class tests carries 20 Marks. Average will be taken of all class tests.

Regularity and attentiveness: 5 Marks

2. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% of the syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20%-30% weightage, and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	ITPCC302	COMPUTER ORGANIZATION & ARCHITECTURE	04

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Exam (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20+25@	30	50	1.5	2	125

@For continuous assessment of tutorials.

Pre-requisite :

1. ESL103: Programming Laboratory - I (C)

Program Outcomes addressed :

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions
4. PO5: Engineering tool usage
5. PO8: Individual and collaborative team work

Course Objectives :

1. To familiarize learners with organizational and architectural aspects of a computer system.
2. To make the learners comprehend Instruction Set of 8086 microprocessor and use it to develop Assembly Language Programs for a given task.
3. To make the learners study and apply algorithms to perform computer arithmetic operations.
4. To make the learners aware of the hierarchical organization of computer memory and various data transfer techniques in digital computer.
5. To acquaint learners with processor performance improvement using instruction level parallelism and advanced computer architectures.

Module	Details	Hrs
00.	**Course Introduction** This is a foundation course which deals with fundamentals of design and functions of different components of a computer system that will enable hardware engineers to develop high performance hardware designs and system programmers to design better system software's aiming at optimizing resource usage.	**01**
01.	**Introduction to Computer Organization and Architecture** ***Learning Objective:***	**04-06**

	Learner is expected to identify and summarize the operational units of a computer system and their interconnections that realize the design specifications..	
	Contents: Introduction to Computer Organization and Architecture. Basic functional blocks of a computer: CPU, memory, input-output subsystems, control unit. Fetch-Decode-Execute Cycle, Von Neumann model. Performance measure of Computer Architecture, Amdahl's Law.	
	Self-Learning Topics: *Evolution of Computers.* ***Learning Outcomes :*** *A learner will be able to* *LO 1.1: Apply the theory and principles of computer science engineering to distinguish between the organizational and architectural aspects of a computer system. (P.I.-1.4.1)* *LO 1.2: Apply engineering fundamentals to restate Amdahl's Law and compute the speedup achieved for a system with a specified number of processors. (P.I.-1.3.1)* *LO 1.3: Identify functionalities and computing resources by summarizing the operations of various units within a computer system. (P.I.-2.2.2)* *LO 1.4: Identify parameters impacting computer architecture performance by analyzing key design considerations and functionalities. (P.I.-2.1.2)*	
02.	**Architecture, Instruction set and programming of 8086 microprocessor** ***Learning Objective/s:*** *Learner is expected to familiarize with the different components of a processor architecture, comprehend and apply the instruction set of 8086 microprocessor to write programs that run on the microprocessor.*	**08-10**
	Contents: Architecture of 8086 Family, Instruction formats, Instruction Set, Addressing Modes, Assembler Directives, Assembly Language Programming, Stack, Procedure, Macro.	
	Self-Learning Topics: *8086 Hardware Design, Minimum and Maximum Mode of operation.* ***Learning Outcomes :***	
	A learner will be able to *LO 2.1: Demonstrate the ability to explore design alternatives by differentiating between procedures and macros in assembly language programming. (P.I.-3.2.1)* *LO 2.2: Produce potential design solutions by applying the instruction set of 8086 microprocessors to develop assembly language programs for a given task. (P.I.-3.2.2)* *LO 2.3: Identify modern engineering tools and techniques for developing assembly language programs. (P.I.-5.1.1)* *LO 2.4: Demonstrate proficiency in using the TASM tool to write and debug assembly language programs. (P.I.-5.2.2)*	

	LO 2.5: Demonstrate effective communication and problem-solving skills by presenting solutions to a given problem in a structured manner. (P.I.-8.2.1) *LO 2.6: Collaborate in teams or individually to solve given problems and effectively illustrate the solutions through integrated efforts. (P.I.-8.3.1)*	
03.	**Processor Organization and Architecture** ***Learning Objective:*** *Learner is expected to analyze different approaches to the design of control unit of processor*	**05-07**
	Contents: CPU Architecture, basic instruction cycle with Interrupt processing. Instruction interpretation and sequencing. Control Unit: Soft wired (Microprogrammed) and hardwired control unit design methods. Microinstruction sequencing and execution. Micro operations, concepts of nanoprogramming.	
	Self-Learning Topics: *Architecture of Pentium IV processor.* ***Learning Outcomes :*** *A learner will be able to* *LO 3.1: Apply engineering fundamentals to summarize the concept and advantages of nano-programming. (P.I.-1.3.1)* *LO 3.2: Apply the theory and principles of computer science engineering to paraphrase the steps taken by a processor during instruction execution. (P.I.-1.4.1)* *LO 3.3: Identify functionalities and computing resources by outlining hardwired and microprogrammed control unit design approaches. (P.I.-2.2.2)* *LO 3.4: Compare and contrast alternative design approaches by analyzing the differences between hardwired and microprogrammed control units. (P.I.-2.2.4)*	
04.	**Data Representation and Arithmetic Algorithms** ***Learning Objective:*** *Learner is expected to familiarize with standard floating point data representation formats and apply Algorithms for performing multiplication and division of integers.*	**05-07**
	Contents: Operations on integers: Negation, Addition, subtraction, Multiplication using Booth's algorithm. Division of integers: Restoring and non-restoring division. Floating-point number representation: IEEE 754 (Single & double precision) floating point number representation.	
	Self-Learning Topics: *Implement Booth's Algorithm and Division methods.* ***Learning Outcomes :*** *A learner will be able to* *LO 4.1: Apply engineering fundamentals to summarize the concept of normalized significand and biased exponent with suitable examples. (P.I.-1.3.1)*	

	LO 4.2: Apply the theory and principles of computer science engineering to restate IEEE 754 floating-point number representation formats. (P.I.-1.4.1) *LO 4.3: Identify functionalities and computing resources by paraphrasing how Booth's algorithm performs multiplication of signed integers. (P.I.-2.2.2)* *LO 4.4: Identify existing methods and apply Booth's algorithm to perform multiplication of signed integers. (P.I.-2.2.3)*	
05.	**Memory and I/O Organization** ***Learning Objective/s:*** *Learner is expected to be aware of the different levels of computer memory to make effective utilization of hierarchical organization of computer memory for faster processing. Also comprehend different data transfer techniques.*	**10-12**
	Contents: Memory Organization: Memory parameters, Classifications of memories, Types of RAM and ROM, Memory hierarchy. Cache memory: Concept and Organization, Elements of a cache design, mapping techniques, Cache write policies, Cache Coherency, High speed memories: Interleaved and Associative memory. I/O Organization: Input/output systems, Structure of peripheral devices, data buffering, I/O module- need, functions and structure, Types of data transfer techniques: Programmed I/O, Interrupt driven I/O and DMA .	
	Self-Learning Topics: *Case study on Memory Organization, Address mapping, Comparison of all I/O methods.* ***Learning Outcomes :*** *A learner will be able to* *LO 5.1:Apply engineering fundamentals to restate the necessity of an Input Output module for efficient data handling in computer systems. (P.I.-1.3.1)* *LO 5.2:Apply theory and principles of computer science engineering to summarize the hierarchical structure of computer memory, highlighting its various levels and access speeds. (P.I.-1.4.1)* *LO 5.3:Identify functionalities and computing resources by paraphrasing the role of a Direct Memory Access (DMA) module in transferring data between peripherals and memory without CPU intervention. (P.I.-2.2.2)* *LO 5.4:Compare and contrast alternative data transfer techniques such as programmed I/O, interrupt-driven I/O, and DMA to select the most efficient method for data movement. (P.I.-2.2.4)*	
06.	**Fundamentals of Advanced Computer Architecture** ***Learning Objective/s:*** *Learner is expected to be aware of the advancements in computer hardware design and techniques used for enhancing the performance of a processor.*	**05-07**
	Contents:	

	Parallel Architecture: Introduction to parallel processing concepts, Flynn's classifications, instruction pipelining, pipeline hazards. Introduction to Array Processors, Clusters, and NUMA Computers. Introduction to Multiprocessor, Multi-Core and Many-Core Systems.	
	Self-Learning Topics: *Advance accelerators such as GPGPUs.* ***Learning Outcomes :*** *A learner will be able to* *LO 6.1:Apply engineering fundamentals to analyze how instruction pipelining enhances processor performance. (P.I.- 1.3.1)* *LO 6.2:Apply the theory and principles of computer science engineering to paraphrase Flynn's classification of parallel computers. (P.I.- 1.4.1)* *LO 6.3:Identify functionalities and computing resources to interpret various pipeline hazards and the methods to mitigate them. (P.I.- 2.2.2)* *LO 6.4:Identify existing solutions and methods to summarize how advanced parallel computer architectures enhance processor performance. (P.I.- 2.2.3)*	
	Course Conclusion	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem
2.2.2	Identifies functionalities and computing resources.
2.2.3	Identify existing solution/methods to solve the problem, including forming justified approximations and assumptions
2.2.4	Compare and contrast alternative solution/methods to select the best methods
2.4.1	Applies engineering mathematics to implement the solution.
3.2.1	Ability to explore design alternatives.
3.2.2	Ability to produce a variety of potential design solutions suited to meet functional requirements.
5.1.1	Identify modern engineering tools, techniques and resources for engineering activities.
5.2.2	Demonstrate proficiency in using discipline specific tools.
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts

Course Outcomes :

Learner will be able to

1. Restate the basic organization and architecture of a computer. *(LO 1.1,LO 1.2, LO 1.3,LO 1.4)*

2. Develop assembly language programs for given task for 8086 microprocessor by using the instruction set of 8086 architecture. *(LO 2.1,LO 2.2, LO 2.3,LO 2.4, LO 2.5,LO 2.6,LO 3.1,LO 3.2, LO 3.3,LO 3.4)*
3. Apply the algorithms to perform computer arithmetic operations. *(LO 4.1,LO 4.2, LO 4.3,LO 4.4)*
4. Analyse the organization of computer memory and different techniques for I/O data transfer.(*LO 5.1,LO 5.2, LO 5.3,LO 5.4)*
5. Summarize advanced computer architectures including performance enhancement using instruction level parallelism. (*LO 6.1,LO 6.2, LO 6.3,LO 6.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC302.1	3	3	-	-	-	-	-	-	-	-	-
ITPCC302.2	3	3	3	-	3	-	-	3	-	-	-
ITPCC302.3	3	3	-	-	-	-	-	-	-	-	-
ITPCC302.4	3	3	-	-	-	-	-	-	-	-	-
ITPCC302.5	3	3	-	-	-	-	-	-	-	-	-
Average	**3**	**3**	**3**	**-**	**3**	**-**	**-**	**3**	**-**	**-**	**-**

Text Books :

1. Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Fifth Edition, 2002, Tata McGraw-Hill.
2. Computer Organization and Architecture: Designing for Performance, William Stallings, Eleventh Edition, 2022, Pearson.
3. 8086/8088 family: Design Programming and Interfacing, John Uffenbeck, First Edition, 1986, Pearson.
4. Advanced Computer Architecture, Smruti R. Sarangi, First Edition, 2021, McGraw Hill.

Reference Books :

1. Computer Architecture and Organization, John P. Hayes, Third Edition, 2017, McGraw-Hill Education.
2. Computer Architecture and Organization: Design Principles and Applications, B. Govindarajulu, Second Edition, 2017, McGraw-Hill Education.
3. Advanced Microprocessors & Peripherals, K Bhurchandi, Third Edition, 2017, McGraw-Hill Education.
4. Advanced Computer Architecture: Parallelism, Scalability, Programmability, Kai Hwang, Naresh Jotwani, Third Edition, 2017, McGraw Hill Education.

Other Resources :

1. NPTEL Course: Computer architecture and organization By Prof. Indranil Sengupta, Prof. Kamalika Datta, Department of Computer Science and Engineering, IIT Kharagpur
   Web link- https://nptel.ac.in/courses/106105163
2. NPTEL Course: Computer Architecture By Prof. Smruti Ranjan Sarangi, Department of Computer Science and Engineering, IIT Delhi
   Web link-https://onlinecourses.nptel.ac.in/noc23_cs67/
3. NPTEL Course: Advanced Computer Architecture By Prof. Smruti Ranjan Sarangi, Department of Computer Science and Engineering, IIT Delhi
   Web link- https://onlinecourses.nptel.ac.in/noc23_cs07/

A. IN-SEMESTER ASSESSMENT (75 MARKS)

1. Continuous Assessment-Theory (20 Marks)

Suggested breakup of distribution

a) One MCQ test as per GATE exam pattern / level: 05 Marks

b) One Class test: 05 Marks

c) One Think Pair Share (TPS) activity: 05 Marks

d) Regularity and active participation: 05 Marks

2. Continuous Assessment - Tutorial (25 Marks)

Suggested breakup of distribution

a) Tutorial Assignments: 20 Marks

Students must be encouraged to write at least 8 class tutorials on entire syllabus in the form of Numerical Assignment, Case Study, Theory Assignment, Programming assignment. Each tutorial assignment carries 20 Marks. Average will be taken of all tutorial assignments.

b) Regularity and active participation: 05 Marks

3. Mid Semester Examination (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20% to 30% weightage, and the syllabus covered from MSE to ESE carrying 70% to 80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	ITPCC303	DATA STRUCTURE AND ANALYSIS	3

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Exam (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite:

1. ESL103: Programming Laboratory -I(C)

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis

Course Objectives:

1. Facilitate the way students, comprehend computer automation in addressing engineering difficulties through the use of data structures.
2. Intended to guide learners through developing complex data structures through the application of fundamental programming concepts.
3. Aid learners to learn the basics of basic data structures including linked lists, stacks, and queues.
4. Assist learners in acquiring a strong comprehension of the fundamental concepts of different types of trees and graphs.
5. Asist learners with understanding the concepts of searching, hashing, and sorting.
6. Assist learners in applying linked lists, stacks, queues, trees, and graphs to solve complicated issues.

Module	Detailed Contents	Hrs
00.	**Course Introduction** This course deals with basics of data types and variables, revision of concept of array, structures and pointers. Data is simplified by data structures. Data structures organize information for machines and humans to understand, most importantly. Understanding DSA helps to solve real-world problems quickly. Data structures and algorithms support logical problem-solving.	01
01.	**Introduction of Data Structures and Algorithms Analysis** ***Learning Objective:*** *Learner is expected to recall and interpret data structures and algorithms to solve real time-complex issues and maximize results.*	07-09
	Contents:	
	Introduction of Data Structures: Concept of computation, algorithms, elementary data types, abstract data types and data structures. RAM model of computation. Data Management concepts, Data types – primitive and non-primitive, Types of Data Structures- Linear & Non-Linear Data Structure. Introduction of Algorithm Analysis: Algorithm, pseudo code for expressing algorithms, performance analysis-space complexity, time complexity, asymptotic notation- big (O) notation, omega notation, theta notation and little (o) notation, recurrences, probabilistic analysis, disjoint set operations, union and find algorithms. applications-analysis of binary search.	
	Self-Learning Topics: *Asymptotic Notations, advanced data structures.* ***Learning Outcomes:*** *A learner will be able to* *LO 1.1: Study the asymptotic notations and apply to analyse algorithm performance for various problem statements. (P.I.-1.3.1)* *LO 1.2: Use fundamentals of data structure and apply its mechanism on computer system and programs. (P.I.-1.4.1)* *LO 1.3: Identify the suitable data structure design for given problem statement. (P.I.-2.3.2)* *LO 1.4: Identify the advantages and disadvantages of searching algorithms by applying on sample data. (P.I.-2.4.3)*	

02.	**Linear Data Structure-Stack, Queue, Linked List** ***Learning Objective:*** *Learner is expected to know linear data structures and interpret all linear data and functions.*	**09-11**
	Contents: Array: Representation of arrays, Applications of arrays, sparse matrix and its representation, Stack: Stack-Definitions & Concepts, Operations on Stacks, Applications of Stacks, Polish Expression, Reverse Polish Expression and Their Compilation, Recursion, Tower of Hanoi, Queue: Representation Of Queue, Operations on Queue, Circular Queue, Priority Queue, Array representation of Priority Queue, Double Ended Queue, Applications of Queue. Linked List: Singly Linked List, Doubly Linked list, Circular linked list, Linked implementation of Stack, Linked implementation of Queue, Applications of linked list.	
	Self-Learning Topics: *Linked List Implementation of Stack, Linked List implementation of Queue, Circular Queue, Double Ended Queue, Priority Queue.* ***Learning Outcomes:*** *A learner will be able to* *LO 2.1: Discover the various applications by applying mechanism of queues, linked lists, and stacks. (P.I.- 1.3.1)* *LO 2.2: Apply theory of data structure functions to build various application. (P.I.-1.4.1)* *LO 2.3: Select appropriate data structure for various applications. (P.I.-2.1.2)* *LO 2.4: Identify constraints of static data structure design over dynamic data structures. (P.I.-2.3.2)*	

03.	**Nonlinear Data Structure-Tree, Graph** ***Learning Objective:*** *Learner is expected to know and interpret non-linear data structures, utilize them, and analyse its many functions.*	**10-12**

	Contents: Tree-Definitions and Concepts, Representation of binary tree, Binary tree traversal (In-order, Post-order, Preorder), Threaded binary tree, Binary search trees, Conversion of General Trees to Binary Trees, AVL trees, Height Balanced, Weight Balance. Insertion and deletion cases of tree. Graph-Types of Graphs, Matrix Representation of Graphs, Elementary Graph operations, (Breadth First Search, Depth First Search, Spanning Trees, Shortest path, Minimal spanning tree).	
	Self-Learning Topics: *Implementation of AVL and B+ Tree.* ***Learning Outcomes:*** *A learner will be able to* *LO 3.1: Design MST. Learn MST designing using Prim's and Kruskal's algorithm. (P.I.-1.3.1)* *LO 3.2: Apply non-linear data structure mechanism to construct given type of tree. (P.I.-1.4.1)* *LO 3.3: Apply Prim's and Kruskal algorithm to design MST and compare results. (P.I.-2.3.2)* *LO 3.4: Identify the limitations of BST and use of AVL tree. (P.I.-2.4.3)*	
04.	**Sorting and searching** ***Learning Objective:*** *Learner is expected to recall searching and sorting algorithm. Also expected to apply it to order the data with list and arrays.*	**03-05**
	Contents: Introduction of Sorting methods, Concept, Insertion Sort, Quick Sort, Merge Sort, Selection Sort, Heap Sort, Sorting on Several Keys, List and Table Sort. Introduction of Searching concept, Linear Search, Binary Search, Comparison and Complexity measures.	
	Self-Learning Topics: *Implementation of different sorting techniques and searching.* ***Learning Outcomes:*** *A learner will be able to* *LO 4.1: Explore sorting methods on given data values. (P.I.-1.3.1)* *LO 4.2: Compare searching methods. (P.I.-1.4.1)* *LO 4.3: Compare between sorting techniques and its time complexities. (P.I.-2.3.2)* *LO 4.4: Apply sorting and compare result with another method. (P.I.-2.4.3)*	

05.	**Hashing and File Structures** *Learning Objective:* *Learner is expected to recall and summarize understand hashing and file structures to simplify the process of finding or using the original string.*	**05-07**
	Contents: Hashing: The symbol table, Hashing Functions, Collision-Resolution Techniques, File Structure: Concepts of fields, records and files, Sequential, Indexed and Relative/Random File. Organization, indexing structure for index files, hashing for direct files, multi-key file organization and access methods.	
	Self-Learning Topics: *Implementation of applications of hashing methods and collision techniques.* ***Learning Outcomes:*** *A learner will be able to* *LO 5.1: Apply collision handling methods in hashing. (P.I.-1.3.1)* *LO 5.2: Apply Hash function on given datasets and find suitable solution. (P.I.-1.4.1)* *LO 5.3: Apply and compare various collision handling methods in hashing. (P.I.-2.3.2)* *LO 5.4: Identify limitations of searching and benefit of hashing. (P.I.-2.4.3)*	
06.	**Data Structures for Complex Problems of computations** ***Learning Objective/s:*** *Learner is expected to know and recall the use of data structure for solving and analysing complex problems.*	**03-05**
	Contents: Single-source shortest path computation, topological sorting of a partially ordered set. Convex- hull computation, string matching algorithms, median computation, distributed algorithms.	
	Self-Learning Topics: *Implementation of applications for Stack, Queues, Linked List, Trees and Graph.* ***Learning Outcomes:*** *A learner will be able to* *LO 6.1: Choose data structure to solve topology sort of partially ordered set. (P.I.-1.3.1)* *LO 6.2: Decide best suitable data structure for convex hull computation. (P.I.-1.4.1)* *LO 6.3: Apply various data structure solutions for given problem statement. (P.I.-2.3.2)* *LO 6.4: Analyse topology sorting by using different data structures. (P.I.-2.4.3)*	
	Course Conclusion	**01**

Total	**45**

Performance Indicators:

P.I. No.	**P.I. Statement**
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem.
2.3.2	Identify design constraints for required performance criteria.
2.4.3	Identify the limitations of the solution and sources/causes.

Course Outcomes:

Learner will be able to

1. Determine the principles of algorithms and data structures. *(LO 1.1,LO 1.2, LO 1.3,LO 1.4)*
2. Apply theory of linear data structure to solve the real-life problems. *(LO 2.1,LO 2.2, LO 2.3,LO 2.4, LO 3.1,LO 3.2, LO 3.3,LO 3.4)*
3. Apply the methods of non-linear data structures in real life problem solving. *(LO 4.1,LO 4.2, LO 4.3,LO 4.4)*
4. Analyse the concepts of sorting, searching and hashing techniques in real life problem solving. (*LO 5.1,LO 5.2, LO 5.3,LO 5.4)*
5. Apply and compare functions of linear and non-linear data structure for complex problem solution. (*LO 6.1,LO 6.2, LO 6.3,LO 6.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC303.1	3	3	-	-	-	-	-	-	-	-	-
ITPCC303.2	3	3	3	-	-	-	-	-	-	-	-
ITPCC303.3	3	3	3	-	-	-	-	-	-	-	-
ITPCC303.4	3	3	3	-	-	-	-	-	-	-	-
ITPCC303.5	3	3	3	-	-	-	-	-	-	-	-
Average	**3**	**3**	**3**	-	-	-	-	-	-	-	-

Text Books:

1. Data Structures through C in Depth, S. K Srivastava, Deepali Srivastava, 5th Edition, 2011, BPB Publications.
2. Data Structures Using C by Aaron M Tenenbaum, 1st Edition ,2018, Pearson India.
3. Data Structures using C, Reema Thareja, 2nd Edition, 2011, Oxford.

Reference Books:

1. Fundamentals of Data Structures, Ellis Horowitz, Sartaj Sahni, 5th Edition, 2010, Galgotia Publications.
2. An introduction to data structures with applications, Jean Paul Tremblay, Paul G. Sorenson; 3rd Edition, 1984, Tata McGraw-Hill.
3. Data Structures using C and C++, Rajesh K. Shukla, 2nd Edition, 2009, Wiley India.

Other Resources:

1. Digital material:
   Web Link-Complete DS Data Structure in one shot | Semester Exam | Hindi (youtube.com)
2. Web Link-https://topperworld.in/dsa-handwritten-notes/

A. IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

a) Numerical Assignment/s (Minimum 20 problems) : 05 Marks
b) Class Test based on above numerical assignment: 05 Marks
c) One Flip classroom activity: 05 Marks
d) Regularity and active participation :05 Marks

2. Mid Semester Examination (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20% to 30% weightage, and the syllabus covered from MSE to ESE carrying 70% to 80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	ITPCC304	DATABASE MANAGEMENT SYSTEM	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Examination (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite:

1. ESL 103: Programming Laboratory-I (C)

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions
4. PO8: Individual and Collaborative team work
5 PO9: Communication
6 PO11: Life-long learning

Course Objectives:

1. To impart database concepts, including database structures, levels of abstraction, ER modeling, EER modeling, and relational schema design for real-world applications
2. To introduce SQL commands for data definition (DDL), data manipulation (DML), integrity constraints, security management (DCL), indexing, and query optimization for effective database management.
3. To impart Knowledge **of** functional dependencies, schema decomposition, and normalization techniques (1NF, 2NF, 3NF, BCNF) to enhance database efficiency and integrity.
4. To acquaint knowledge of database transaction management, ACID properties, concurrency control mechanisms, deadlock handling, and recovery techniques to ensure data consistency and security.
5. Impart knowledge to students on working collaboratively in teams, communicating database design decisions, and applying security, indexing, and optimization techniques to real-world database solutions

Module	Details	Hrs.
00.	**Course Introduction**	**01**
01.	**Introduction to DBMS** ***Learning Objective:*** ***To impart Learner to recall the concept of database system architecture and interpret the process of data management***	**4-5**

	Contents: Introduction, Difference between data and information, Characteristics of Databases, File system v/s Database system, Data abstraction and Physical Data Independence, DBMS system architecture, Database Administrator (DBA), Role of DBA, significance of Relational Database	
	Self-Learning Topics: ***Identify the types of Databases.***	
	Learning Outcomes: *A learner will be able to* *LO 1.1: Summarize the key characteristics of databases and compare them with file systems. Apply engineering principles to evaluate the advantages of databases, including data integrity, security, scalability, and efficiency in data management. (PI- 1.3.1)* *LO 1.2: Identify and categorize different levels of abstraction in database systems. Apply a structured approach to analyze their roles in data organization, access, and management. (PI- 2.1.1)* *LO 1.3: Draw and explain the structure of a database system using engineering and computer science principles. Analyze the components and their interactions to understand database functionality and management.* (*PI-1.4.1)* *LO 1.4: Apply the principles of physical data independence to ensure flexibility in database management. Analyze how changes in physical storage structures do not affect logical data organization and application programs. (PI-2.1.2)*	
02.	**E-R model** ***Learning Objective:*** ***To introduce learner recall and demonstrate the concept of ER model. Also expected to apply the concept of ER model for conceptual design of database***	**5-7**
	Contents: Conceptual Modeling of a database, The Entity-Relationship (ER) Model, Entity Type, Entity Sets, Attributes and Keys, Relationship Types, Relationship Sets, Weak entity Types. Extended Entity-Relationship (EER) Model: Generalization, Specialization and Aggregation. Design an ER model for real time case study. (e.g. Hospital management system)	
	Self-Learning Topics: ***Design an ER model for any real time case study***	
	Learning Outcomes: *A learner will be able to* *LO 2.1: Recognize and categorize entity sets and relationship sets based on their characteristics and functionalities. Apply a structured approach to represent relationships effectively in database design. (PI-2.1.1)* *LO 2.2: Implement constraints with mapping cardinality to define relationships between entities. Ensure data integrity and consistency by applying structured approaches to enforce database constraints. (PI- 2.1.2)* *LO 2.3: Develop Entity-Relationship (ER) diagrams to model real-time problems. Apply structured engineering approaches to accurately represent entities, attributes, relationships, and constraints in database design. (PI- 3.1.1)* *LO 2.4: Design Enhanced Entity-Relationship (EER) diagrams to model complex real-time problems. Apply structured approaches to represent specialization, generalization, and inheritance in database systems. (PI- 3.1.2)*	

	LO 2.5: Develop conceptual database models using the Entity-Relationship (ER) approach. Identify entity types, attributes, keys, and relationship sets, including weak entities, to ensure structured data representation. (PI- 1.3.1) *LO 2.6: Create ER and Extended ER (EER) models incorporating generalization, specialization, and aggregation. Apply engineering principles to design a comprehensive database model for real-time case studies, such as a hospital management system. (PI- 1.4.1)*	
03.	**Introduction to Relational Model** ***Learning Objective:*** *To impart learner recall and demonstrate the process of conversion of ER into relational model. Also expected apply the process to create a database.*	**5-7**
	Contents: Relational Model Constraints and Relational Database Schemas, Concept of Keys: Primary Kay, Secondary key, Foreign Key, Introduction to Relational Algebra, Relational Algebra expressions for Unary Relational Operations, • Set Theory operations, • Binary Relational operation Relational Algebra Queries, Conversion of ER model in to relational schema by using rules. (e.g. Hospital management system)	
	Self-Learning Topics: *Convert the ER model designed to relational schema by using rules* for real time problem	
	Learning Outcomes: *A learner will be able to* *LO 3.1: Create conceptual database models using the Entity-Relationship (ER) approach. Identify and classify entity types, entity sets, attributes, keys, and relationship types to ensure structured data representation. (PI-1.3.1)* *LO 3.2: Identify and interpret various relationship types, including weak entity types, using engineering and computer science principles. Ensure proper representation of data relationships in database models. (PI- 1.4.1)* *LO 3.3: Analyze and categorize entity sets and relationship sets based on their characteristics and functionalities. Apply a structured approach to effectively represent relationships in conceptual database design. (PI- 2.1.1)* *LO 3.4: Apply integrity constraints and mapping cardinality to define relationships between entities in ER models. Ensure data consistency and accurate representation of business rules in databases. (PI- 2.1.2)* *LO 3.5: Develop ER models to solve real-world problems, such as a hospital management system. Apply structured techniques to accurately model data, relationships, and constraints. Collaborate effectively in teams, communicate database design decisions clearly, and utilize modern tools to enhance problem-solving in database modeling. (PI- 3.1.1, 8.1.1, 9.1.1, 11.2.1)* *LO3.6: Enhance ER models by incorporating Extended Entity-Relationship (EER) concepts such as generalization, specialization, and aggregation. Apply advanced database modeling techniques to design comprehensive real-time systems. Collaborate effectively in teams, communicate design decisions clearly, and stay updated with evolving database technologies to ensure efficient system implementation. (PI- 3.1.2, 8.1.2, 9.1.3, 11.2.2)*	
04.	**Overview of SQL** ***Learning Objectives:*** *To introduce learner recall and illustrate SQL and also expected to build Queries*	**11-12**

	Contents: Overview of SQL, Data Definition Commands, set operations, aggregate function, null values Data Definition language(DDL), Data Manipulation commands(DML), Data Control commands(DCL), Complex Retrieval Queries using Group by, Recursive Queries, nested Queries, Integrity constraints in SQL, Security and authorization: Grant & Revoke in SQL. Indexing: Basic Concepts, Ordered Indices, Index Definition in SQL Physical design of database for the relational model designed (e.g. Hospital management system)	
	Self-Learning Topics:	
	Learning Outcomes: *A learner will be able to* *LO 4.1: Understand and use SQL for defining and managing data. Apply Data Definition Language (DDL) and Data Manipulation Language (DML) commands to create, modify, and manipulate relational databases efficiently. (PI- 1.3.1)* *LO 4.2: Execute SQL set operations, aggregate functions, and handle null values effectively. Apply structured approaches to ensure data integrity and meaningful query results. (PI-1.4.1)* *LO 4.3: Develop advanced SQL queries, including GROUP BY, recursive queries, and nested queries, to retrieve and analyze relational data efficiently. (PI-2.1.1)* *LO 4.4: Implement and enforce integrity constraints in SQL to maintain data consistency and accuracy in relational databases. (PI- 2.1.2)* *LO 4.5: Use GRANT and REVOKE commands to manage security and access control in SQL databases. Ensure proper user role management, data protection, and secure multi-user access. Collaborate effectively to enforce security policies, communicate security measures clearly, and apply best practices for database protection. (PI- 3.1.1, 8.2.1, 9.1.1, 11.2.1)* *LO 4.6: Apply indexing techniques such as ordered indices and define indexes in SQL to enhance database performance. Design the physical structure of relational databases to optimize query execution and storage management. Utilize problem-solving approaches, collaborate effectively in teams, and stay updated with advanced database optimization techniques. (PI- 3.1.2, 8.3.1, 9.1.3, 11.2.2)*	
05.	**Normalization** ***Learning Objective/s:*** *To introduce learner define and interpret the concept of normalization and also expected to convert the given relation in different normalization forms to remove redundancy.*	**6-8**
	Contents: Design guidelines for relational Schema, Functional Dependencies, Database tables and normalization, the need for normalization, the normalization process, Improving the design, Definition of Normal Forms- 1NF, 2NF, 3NF & The Boyce-Codd Normal Form (BCNF).	
	Self-Learning Topics: *Consider any real time application and apply normalization up to 3NF/BCNF to remove anomalies.*	
	Learning Outcomes :	
	A learner will be able to	

	LO 5.1: Implement best practices for designing relational schemas to ensure data consistency, integrity, and efficiency in database management. (PI-1.3.1) *LO 5.2: Identify and evaluate functional dependencies to establish relationships between attributes and enhance database structure.* (PI: 1.4.1) *LO 5.3: Understand the need for normalization and apply normalization techniques to enhance data organization and eliminate redundancy in relational databases.* (PI: 2.1.1) *LO 5.4: Apply structured methodologies to decompose relational tables and improve database design through normalization.* (PI: 2.1.2) *LO 5.5: Explain the characteristics of First Normal Form (1NF), Second Normal Form (2NF), and Third Normal Form (3NF) and apply them to refine relational schemas for efficient data storage and retrieval. Collaborate effectively in teams, communicate database design improvements clearly, and apply best practices to ensure data integrity and system efficiency. (PI- 3.1.1, 8.2.1, 9.1.1, 11.2.1)* *LO 5.6: Implement Boyce-Codd Normal Form (BCNF) to eliminate anomalies and enhance database structures. Ensure higher data integrity and efficiency by applying advanced normalization techniques. Collaborate effectively, communicate database optimizations clearly, and stay updated with best practices in database design. (PI- 3.1.2, 8.3.1, 9.1.3, 11.2.2)*	
06.	**Transaction** ***Learning Objective/s:*** *To introduce learner define and illustrate transaction concept, Concurrency Control and Recovery. Also expected to apply the concept to ensure database consistency.* **Contents:** Transaction concept, State Diagram, ACID Properties, Transaction Control Commands(TCL), Concurrent Executions, Serializability – Conflict and View, Concurrency Control: Lock-based-protocols, Deadlock handling Timestamp-based protocols, Recovery System: Shadow Paging Recovery Concepts, Log based recovery. ***Self-Learning Topics:*** *Study the various deadlock situation which may occur for a database designed in module 5* ***Learning Outcomes:*** *A learner will be able to* *LO 6.1: Interpret the state diagram of transactions and apply Transaction Control Commands (TCL) such as COMMIT, ROLLBACK, and SAVEPOINT to manage database transactions effectively. (PI-1.4.1)* *LO 6.2: Interpret the state diagram of transactions and apply Transaction Control Commands (TCL) such as COMMIT, ROLLBACK, and SAVEPOINT to manage database transactions effectively.* (PI: 1.4.1) *LO 6.3: Analyze concurrent executions in database systems and apply conflict and view serializability techniques to maintain data consistency in multi-user environments. (PI-2.1.1)* *LO 6.4: Implement lock-based protocols, timestamp-based protocols, and deadlock handling strategies to ensure smooth and efficient execution of concurrent transactions. (PI- 2.1.2)*	**6-8**
	Course Conclusion	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply fundamental engineering concepts to solve engineering problems.
1.4.1	Apply electrical engineering concepts to solve engineering problems.
2.1.1	Evaluate problem statements and identifies objectives
2.3.2	Identify design constraints for required performance criteria.
3.2.1	Ability to explore design alternatives.
3.2.2	Ability to produce a variety of potential design solutions suited to meet functional requirements
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts
9.1.1	Read, understand and interpret technical and nontechnical information
9.1.3	Create flow in a document or presentation - a logical progression of ideas so that the main point is clear
11.2.1	Identify historic points of technological advance in engineering that required practitioners to seek education in order to stay current
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field

Course Outcomes: A learner will be able to -

1. Apply engineering principles to create ER and EER models, define entity relationships, and optimize database schemas for real-world applications. *(LO 1.1, 1.2, 1.3, 1.4, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 3.1, 3.2, 3.3, 3.4)*
2. Demonstrate proficiency in SQL queries, integrity constraints, security commands (GRANT, REVOKE), indexing, and query optimization to efficiently manage databases. *(LO 4.1, 4.2, 4.3, 4.4, 4.5, 4.6)*
3. Evaluate functional dependencies and apply normalization techniques (1NF, 2NF, 3NF, BCNF) to refine relational schemas and eliminate redundancy. *(LO 5.1, 5.2, 5.3, 5.4, 5.5, 5.6)*
4. Write TCL commands, serializability techniques, concurrency control protocols, deadlock handling, and recovery mechanisms to manage transactions securely. *(LO 6.1, 6.2, 6.3, 6.4)*
5. Work collaboratively on database projects, apply security best practices, optimize query execution, and improve database performance using indexing and normalization. *(LO 3.5, 3.6, 4.5, 4.6, 5.5, 5.6, 6.3, 6.4)*

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC304.1	3	3	3	-	-	-	-	-	-	-	-
ITPCC304.2	3	3	3	-	-	-	-	3	3	-	3
ITPCC304.3	3	3	3	-	-	-	-	3	3	-	3
ITPCC304.4	3	3	-	-	-	-	-	-	-	-	-
ITPCC304.5	-	3	3	-	-	-	-	3	3	-	3
Average	**3**	**3**	**3**	-	-	-	-	**3**	**3**	-	**3**

Text Books :

1. Database System Concepts, Korth, Slberchatz, Sudarshan , Sixth Edition, 2011, McGraw Hill Education
2. Fundamentals of Database Systems, Elmasri and Navathe, Sixth Edition, 2010, Pearson
3. Database Management Systems, Raghu Ramkrishnan and Johannes Gehrke, ISE Editions, 1997 ,McGraw-Hill Education

Reference Books :

1. Database Systems Design, Implementation and Management , Peter Rob and Carlos Coronel, 9th Edition, 2009, Thomson Learning
2. SQL & PL / SQL for Oracle 11g Black Book, Dr. P. S. Deshpande, ISE Edition, 2011,Dreamtech Press
3. Database Management Systems , G. K. Gupta , 1st edition,2011,McGraw – Hill

Other Resources :

1. NPTEL Course on Database Management system by Prof.P.P.Das, Department of Computer Science and Engineering, IIT Khargpur
   Web link- https://archive.nptel.ac.in/courses/106/105/106105175/
2. Coursera course on Introduction to Relational Databases
   Web link- https://www.coursera.org/learn/introduction-to-relational-databases#modules

A. IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment - Theory-(20 Marks)

Suggested breakup of distribution

a) One Assignment on live problems/ case studies: 10 Marks

Students should be assigned a real life problem statement (different for each student). Students are expected to research and collect required resources to create a backend for the selected problem. Students should prepare a presentation of 10-15 minutes. This assignment should be graded for 10 marks depending on the parameters as analysis, design, conversion to relational schema and database creation for selected problem statement.

b) One Think Pair Share (TPS) activity: 05 Marks

c) Regularity and active participation :05 Marks

1. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20%-30% weightage, and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
LBC	ITLBC301	DATA STRUCTURES LABORATORY	01

Examination Scheme		
Continuous Assessment	**Practical /Oral**	**Total**
25	25	50

Pre-requisite:

1. ESL103: Programming Laboratory –I (C)

Program Outcomes addressed:

1. PO1: Engineering Knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solution

Course Objectives:

1. To introduce learners with the use of data structures and algorithm analysis.
2. To instruct learners about creating complex data structures via basic programming concepts.
3. To train learners about creating realistic implementation of linked lists, stacks, and queues.
4. To guide learners in the actual use of various tree and graph types.
5. To provide knowledge to learners on how to develop searching, hashing, and sorting algorithms.
6. To prepare students to use linked lists, stacks, queues, trees, and graphs to complicated problems.

Module	Detailed Contents	Hrs
00.	**Course Introduction** This course is intended to teach the design and analysis of basic data structures and their implementation in an object-oriented language. The data structures concepts are increasingly becoming the default choice of the IT industry especially industries involved in software development at system level.	**--**

01.	**Implementation of basic data structures** ***Learning Objective/s:*** *Learner is expected to apply unions, pointers, 1D arrays, and structures to solve a given problem.* **Experiment 1:** WPC for demonstration of nested structures using function for student having name, id, marks [3]. Demonstrate the structure for 10 students. **Experiment 2:** Demonstrate pointers and character array.	**04**
	Self-Learning Topics: *Pointer implementation of all integrated data types.* ***Learning Outcomes:*** *A learner will be able to* *LO 1.1: Compile and execute C program on Ubuntu OS using CLI. (P.I.-1.3.1)* *LO 1.2: Code array as 1D and 2D data structure. (P.I.-1.4.1)* *LO 1.3: Implement linear search using 1D-array. (P.I.-2.2.3)* *LO 1.4: Identify the capability of 16bit and 32bit machine to use integer data type. (P.I.-2.3.1)* *LO 1.5: Use 2D array for mathematical matrix operation. (P.I.- 3.4.2)* *LO 1.6: Compare the mythology using array and structures. (P.I.- 3.4.3)*	
02.	**Implementation of Static Linear DS** ***Learning Objective:*** *Learner is expected to apply linear data structures and their application in fixed-size scenarios through the use of arrays.* **Experiment 3:** Write menu driven program to implement static stack with push, pop, peek and display functions of it. **Experiment 4:** Write menu driven program to implement static queue and circular queue of size 10 with insert, delete, peek and display functions of it.	**06**
	Self-Learning Topics: *Implement character stack, stack applications.* ***Learning Outcomes:*** *A learner will be able to* *LO 2.1: Restate the principle of stack and queue data structure. (P.I.-1.3.1)* *LO 2.2: Apply concept of array to implement functions of stack data structure. (P.I.-1.4.1)* *LO 2.3: Use array to design functions of queue data structure. (P.I.-2.2.3)* *LO 2.4: Identify and implement static applications at system level. (P.I.-2.3.1)* *LO 2.5: Implement evaluation and conversion of Infix, prefix, postfix expression. (P.I.- 3.4.2)* *LO 2.6: Define problem and find solution using various data structures. (P.I.- 3.4.3)*	

03.	**Implementation of Linear Data Structure using Dynamic Memory Allocation** ***Learning Objective:*** *Learners is expected to comprehend the primary advantage of linked lists compared to arrays and be proficient in using dynamic memory allocation.* **Experiment 5:** Implementation of singly linked list data structure with given function. **Experiment 6:** Implementation of doubly linked list data structure with given function. ***Self-Learning Topics:*** *Study and implement header linked list.* ***Learning Outcomes:*** *A learner will be able to* *LO 3.1: Use of dynamic data memory allocation for implementation. (P.I.-1.3.1)* *LO 3.2: Illustrate the concept and principle of linked list data structure. (P.I.-1.4.1)* *LO 3.3: Design the LL with various functions and compare with other type. (P.I.-2.2.3)* *LO 3.4: Apply the concept of LL in real time applications. (P.I.-2.3.1)* *LO 3.5: Model the application using LL data structure (P.I.- 3.4.2)* *LO 3.6: Demonstrate the LL for system and kernel level functions. (P.I.- 3.4.3)*	**08**

04.	**Implementation of graph and tree** ***Learning Objective:*** *Learner is expected to understand computer file systems and business organizational chart tree structures and implement graph data structures.* **Experiment 7:** Implement binary tree using insertion and deletion of node. **Experiment 8:** Implement tree traversal for in-order, preorder and post order types. **Experiment 9:** Write program to implement adjacency matrix of graph. **Experiment 10:** Implement BFS and DFS traversal of graph.	**04**

	Self-Learning Topics: *Implement 2-3, and B and B+ tree mechanism and graph algorithms.* ***Learning Outcomes:*** *A Learner will be able to* *LO 4.1: Execute the code for tree data structure. (P.I.-1.3.1)* *LO 4.2: Identify use of 2D array to design graph. (P.I.-1.4.1)* *LO 4.3: Compare BFS and DFS traversal of graph. (P.I.-2.2.3)* *LO 4.4: Identify the real time application of Graph for Map optimization. (P.I.-2.3.1)* *LO 4.5: Explore various shortest path algorithms. (P.I.- 3.4.2)* *LO 4.6: Demonstrate system level functions implementation using tree data structures. (P.I.- 3.4.3)*	
05	**Design and Implementation of searching and sorting** ***Learning Objectives:*** *Learner is expected to apply linear data structures and their application in fixed-size scenarios through the use of arrays.* **Experiment 11:** Write program to implement linear and binary search. **Experiment 12:** Write program using recursion for implementation of quick sort. **Experiment 13:** Write program for implementation of insertion/ selection sort.	**04**
	Self-Learning Topics: *Learn to code shell and bucket sort.* ***Learning Outcomes:*** *A learner will be able to* *LO 5.1: Calculate the time complexity of linear and binary search. (P.I.-1.3.1)* *LO 5.2: Code the searching functions. (P.I.-1.4.1)* *LO 5.3: Identify various sorting mechanisms (P.I.-2.2.3)* *LO 5.4: Apply any two types of sorting methods on given set of data. (P.I.-2.3.1)* *LO 5.5: Determine and analyze time complexity of searching and sorting above methods. (P.I.- 3.4.2)* *LO 5.6: Identify best method of sorting for large set of data. (P.I.- 3.4.3)*	
06.	**Implementation of applications of Data structures**	**04**
	Learning Objective: *Learner is expected to implement complex problems using data structures and algorithms, and analysing data structures inside data management models* **Experiment 14:** WPC using stack data structure for infix to postfix algorithm. **Experiment 15:** Implementation of Josephus Problem using circular linked list. **Experiment 16:** Implementation of hashing functions with different collision resolution techniques.	

	Self-Learning Topics**:* *Study advanced data structures for real time problems.* ***Learning Outcomes: *LO 6.1: Study various complex problem in data management and retrieval. (P.I.-1.3.1)* *LO 6.2: Select the data structure to implement the solution of problem. (P.I.-1.4.1)* *LO 6.3: Analyse the working and use various data structure for real world problem. (P.I.-* *2.2.3)* *LO 6.4: Compare the use of stack and queue for various applications. (P.I.-2.3.1)* *LO 6.5: Compare the solution of specific problem by using more than one data structure. (P.I.- 3.4.2)* *LO 6.6: Model the collision resolution for large set of data and find computational speedup on available resources. (P.I.- 3.4.3)*	
	Total	**30**

Performance Indicators:

P.I. No.	**P.I. Statement**
1.3.1	Apply engineering fundamentals
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem.
2.2.3	Identify existing solution/methods to solve the problem, including forming justified approximations and assumptions.
2.3.1	Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.
3.4.2	Ability to implement and integrate the modules.
3.4.3	Ability to verify the functionalities and validate the design.

Course Outcomes:

Learner will be able to

1. Design and use the basic concepts and principles of various data types. *(LO 1.1,LO 1.2, LO 1.3,LO 1.4, LO 1.5,LO 1.6)*
2. Implement the concepts of the methods of Linear data structures. *(LO 2.1,LO 2.2, LO 2.3,LO 2.4,LO 2.5, LO 2.6, LO 3.1,LO 3.2, LO 3.3,LO 3.4,LO 3.5, LO 3.6)*
3. Design and implement various functions of tree and graph. *(LO 4.1,LO 4.2, LO 4.3,LO 4.4, LO 4.5,LO 4.6)*
4. Design the concepts and compare the techniques of searching, hashing and sorting. *(LO 5.1,LO 5.2, LO 5.3,LO 5.4,LO 5.5,LO 5.6)*
5. Apply and examine the methods of linked lists, stacks, queues, trees and graphs to solve complex problems. *(LO 6.1,LO 6.2, LO 6.3,LO 6.4,LO 6.5,LO 6.6)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITLBC301.1	3	3	3	-	-	-	-	-	-	-	-
ITLBC301.2	3	3	3	-	-	-	-	-	-	-	-
ITLBC301.3	3	3	3	-	-	-	-	-	-	-	-
ITLBC301.4	3	3	3	-	-	-	-	-	-	-	-
ITLBC301.5	3	3	3	-	-	-	-	-	-	-	-
Average	**3**	**3**	**3**	-	-	-	-	-	-	-	-

Text Books:

1. Data Structures through C in Depth, S. K Srivastava, Deepali Srivastava, 5th Edition, 2011, BPB Publications.
2. Data Structures Using C by Aaron M Tenenbaum, 1st Edition 2018, Pearson India.
3. Data Structures using C, Reema Thareja, 2nd Edition, 2011, Oxford.

Reference Books:

1. Fundamentals of Data Structures, Ellis Horowitz, Sartaj Sahni, 5th Edition, 2010, Galgotia Publications.
2. An introduction to data structures with applications, Jean Paul Tremblay, Paul G. Sorenson; 3rd Edition, 1984, Tata McGraw-Hill.
3. Data Structures using C and C++, Rajesh K. Shukla, 2nd Edition, 2009, Wiley India.

Other Resources:

1. Digital material:
   Web Link- Complete DS Data Structure in one shot | Semester Exam | Hindi (youtube.com)
2. Web Link- https://topperworld.in/dsa-handwritten-notes/

A. CONTINUOUS ASSESSMENT (25 MARKS)

Suggested breakup of distribution

a) Task Execution :10 Marks

Students will be given minimum 12 tasks based on list as per mentioned in the syllabus. Each task carries 10 Marks. Average will be taken of all tasks. Students are expected to execute the program for given task with respect to given data structure, also expected to perform

searching & sorting operations on given dataset. Students will be evaluated based on logic building for the given task and expected output.

b) Practical Test: 10 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

c) Regularity & Active participation: 05 Marks

B. END SEMESTER EXAMINATION (Practical and Oral Exam) (25 Marks)

Suggested breakup of distribution

a) Task Execution: 10 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

b) Results and discussion, Inferences drawn based on above task: 05 Marks

c) Oral based on entire syllabus :10 Marks

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
LBC	ITLBC302	SQL LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Exam (ESE)**	**Total**
25	25	50

Pre-requisite:

1. ESL103 : Programming Laboratory-I (C)

Program Outcomes addressed:

1 PO1: Engineering knowledge

2 PO2: Problem analysis

3 PO3: Design/Development of Solutions

4 PO5: Engineering tool usage

5 PO8: Individual and Collaborative team work

6 PO09: Communication

7 PO11: Life-long learning

Course Objectives:

1. To introduce students database modeling concepts, including entity-relationship (ER) and enhanced entity-relationship (EER) diagrams. Train them to establish relationships, apply integrity constraints, and ensure data consistency using engineering principles
2. Familiarize students with SQL commands for creating, modifying, and securing databases. Teach the use of **GRANT**, **REVOKE**, and transaction control commands to enforce access control and optimize database performance.
3. Impart knowledge on implementing views, triggers, and materialized views to automate database operations. Teach structured approaches to enhance database efficiency and performance.
4. To develop designing and implementing real-time database solutions using SQL, functions, procedures, and JDBC connectivity. Develop problem-solving skills by collecting stakeholder requirements and defining problem statements effectively.
5. Foster teamwork by training students to communicate requirements, distribute project tasks, and execute database activities while adhering to deadlines

Module	Details	Hrs.
00.	**Course Introduction** This is foundation course which deals with fundamental concepts of SQL to store, update, remove, search, and retrieve information from the database and it is used in all real time applications for management of data.	**01**
01.	**Introduction to ER and Extended ER diagram** ***Learning Objective:***	**07**

	Learner is expected to gain knowledge of ER, Extended ER diagram to demonstrate conceptual design of database and expected to implement it *Learner is expected to define and interpret rules of conversion of ER into relational model to implement a table in a database. Also expected to analyze it for real time application*	
	Contents: **Notation of ER and Extended ER diagram, Mapping the ER and EER Model to the Relational Model** **Task 1:** Identify real world problem and develop the problem statement. Design an Entity-Relationship (ER) / Extended Entity-Relationship (EER) Model **Task 2:** Mapping ER/EER to Relational schema model for task 1.	
	Self-Learning Topics:	
	Learning Outcomes: *A learner will be able to* *LO 1.1: Apply engineering principles to accurately represent entities, attributes, relationships, and constraints in database modeling. Identify and classify entity and relationship sets based on their characteristics and functionalities. Use primary keys, foreign keys, and mapping cardinality constraints to establish relationships and ensure data integrity in relational databases. (PI- 1.3.1, 2.1.1, 3.2.2)* *LO 1.2: Identify and interpret relationships between entities using engineering and computer science principles to design well-structured database models. Apply structured approaches to create ER and EER diagrams, representing complex entity relationships and hierarchical data. Convert ER diagrams into relational models by defining tables, attributes, keys, and relationships while ensuring normalization and data integrity. (PI- 1.4.1, 2.1.2, 3.1.1)*	
02.	**Types of Language in SQL**	**12**
	Learning Objective: *Learner is expected to list and identify Data Manipulation commands to implement data manipulation in database for real time application* *Learner is expected to define and apply Data Control commands to assign authorization to the database user as per their role and implement it for real time application* *Learner is expected to acquire knowledge and demonstrate retrieval of information as per requirement of organization and implement it for real time application*	
	Contents: **Data Definition Commands, Data Manipulation commands, Data Control commands, Complex SQL queries**	
	Self-Learning Topics:	
	Learning Outcomes:	
	A learner will be able to *LO 2.1: Create tables using SQL while applying integrity constraints to ensure data consistency and accuracy. Perform add, delete, and update operations using SQL commands. Manage database users by defining access controls and permissions. Apply structured approaches to optimize database functionality and performance. (PI- 2.1.1, 2.3.2, 3.1.1, 3.2.2)*	

	*LO 2.2: Apply SQL commands to manage database security and user roles using **GRANT** and **REVOKE** permissions. Execute basic and complex queries to retrieve, update, and manipulate data effectively. Use structured approaches to enforce access control, optimize database performance, and ensure data integrity. (PI- 3.1.2, 5.1.1, 5.1.2, 5.2.2)*	
03.	**Views and Triggers** *Learner is expected to define and demonstrate Views and Triggers to provide access to the required data only and implement it for real time application.*	**02**
	Contents: **Views and Triggers** **Task 7 :** **Implementation of Views and Triggers real time application**	
	Self-Learning Topics:	
	Learning Outcomes: *A learner will be able to* *LO 3.1: Summarize the concept and functionality of views and triggers in database systems. Write the syntax for creating views and develop views for real-time applications. Apply structured approaches to implement triggers for automating database operations and maintaining data integrity.* (PI-2.3.2, 2.1.1, 5.1.1, 3.1.2) *LO 3.2: Write and implement triggers for real-time applications to automate database operations and maintain data consistency. Define materialized views and differentiate them from standard views based on their functionality and performance benefits. Apply structured approaches to enhance database efficiency and optimization.* (PI-3.1.1, 5.1.2, 1.3.1, 1.4.1)	
04.	**TCL and Functions and procedures** ***Learning Objectives:*** ***Learner is expected to define and demonstrate Views and Triggers to provide access to the required data only and implement it for real time application. Learner is expected to define and demonstrate Transaction Control Language to use commit and rollback command to save tables in databases and implement it for real time application***	**07**
	Contents: **Transaction control Language, Functions and procedures, Database Programming , Mini project on real time problem**	
	Self-Learning Topics:	
	Learning Outcomes:	
	A learner will be able to *LO 4.1: Use **COMMIT**, **ROLLBACK**, and **SAVEPOINT** commands to manage* database transactions and ensure data integrity. Summarize the role of functions and procedures in SQL and their importance in automating database operations. Apply structured approaches to enhance database reliability and performance. *(PI- 1.4.1, 1.3.1, 5.1.1, 5.1.2)* LO 4.2: Create functions and procedures in SQL to enhance database automation and efficiency. Demonstrate database connectivity using JDBC for real-time applications. Design and implement real-time database solutions based on problem statements. Collect stakeholder requirements and define problem statements without conflict to ensure effective database system development. *(PI- 3.1.2, 3.1.1, 11.2.2, 11.1.1, 2.3.2, 9.1.1, 8.1.1, 2.1.1, 8.3.1, 9.3.2)* LO 4.3 *Communicate project requirements clearly among team members and allocate tasks based on expertise. Execute assigned activities as per the project plan*	

	while adhering to deadlines. Apply structured approaches to ensure teamwork, efficiency, and successful project completion. Utilize problem-solving skills to optimize workflows and maintain quality standards in database development. (*PI- 9.1.1, 9.3.2, 8.1.1, 8.3.1, 1.3.1, 1.4.1, 2.1.1, 2.3.2, 3.3.3, 3.2.2, 11.1.1, 11.2.2)*	
	Course Conclusion	**01**
	Total	**30**

Performance Indicators:

P.I. No.	**P.I. Statement**
1.3.1	Apply engineering fundamentals
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem
2.1.1	Evaluate problem statements and identifies objectives
2.1.2	Identifies processes/modules/algorithms of a computer based system and parameters to solve a problem
2.3.2	Identify design constraints for required performance criteria.
3.1.1	Able to define a precise problem statement with objectives and scope
3.1.2	Able to identify and document system requirements from stake holders
3.2.2	Ability to produce a variety of potential design solutions suited to meet functional requirements
5.1.1	Identify modern engineering tools, techniques and resources for engineering activities
5.1.2	Create/adapt/modify/extend tools and techniques to solve engineering problems
8.1.1	Recognize a variety of working and learning preferences; appreciate the value of diversity on a team
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts
9.1.1	Read, understand and interpret technical and nontechnical information
9.3.1	Create engineering-standard figures, reports and drawings to complement writing and presentations
11.1.1	Describe the rationale for requirement for continuing professional development
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field

Course Outcomes: A learner will be able to -

1. Apply engineering and computer science principles to create well-structured database models, ensuring proper entity relationships, normalization, and integrity constraints. *(LO 1.1, 1.2)*
2. Create, modify, and secure databases using SQL commands, enforcing constraints, executing queries, and applying transaction control mechanisms. *(LO 2.1, 2.2, 4.1)*
3. Implement views and triggers to automate database operations and define materialized views to optimize data retrieval and storage. *(LO 3.1, 3.2)*
4. Design and implement real-time databases using SQL functions, stored procedures, and JDBC connectivity while considering stakeholder requirements. *(LO 4.2)*

5. Communicate project requirements, distribute tasks efficiently, execute database-related activities as per the project plan, and meet deadlines. *(LO 4.3)*

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITLBC302.1	3	3	3	-	-	-	-	-	-	-	-
ITLBC302.2	3	3	3	-	3	-	-	-	-	-	-
ITLBC302.3	3	3	3	-	3	-	-	-	-	-	-
ITLBC302.4	-	3	3	-	-	-	-	3	3	-	3
ITLBC302.5	-	3	3	-	-	-	-	3	3	-	3
Average	**3**	**3**	**3**	-	**3**	-	-	**3**	**3**	-	**3**

Text Books :

1. Database System Concepts, Korth, Slberchatz, Sudarshan , Sixth Edition, 2011, McGraw Hill Education
2. Fundamentals of Database Systems, Elmasri and Navathe, Sixth Edition, 2010, Pearson
3. Database Management Systems, Raghu Ramkrishnan and Johannes Gehrke, ISE Editions, 1997 ,McGraw-Hill Education

Reference Books :

1. Database Systems Design, Implementation and Management , Peter Rob and Carlos Coronel, 9th Edition, 2009, Thomson Learning
2. SQL & PL / SQL for Oracle 11g Black Book, Dr. P. S. Deshpande, ISE Edition, 2011,Dreamtech Press
3. Database Management Systems , G. K. Gupta , 1st edition,2011,McGraw – Hill

Other Resources :

1. NPTEL Course on Database Management system by Prof.P.P.Das, Department of Computer Science and Engineering, IIT Khargpur
   Web link- https://archive.nptel.ac.in/courses/106/105/106105175/
2. Coursera course on Introduction to Relational Databases
   Web link- https://www.coursera.org/learn/introduction-to-relational-databases#modules

IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment - Theory-(25 Marks)

Suggested breakup of distribution

a) Task Execution :10 Marks

Students will be given 10 tasks based on list as per mentioned in the syllabus. Each task carries 10 Marks. Average will be taken of all tasks. Students are expected to formulate problem statement, create relational schema, implement relational schema by using constraints and retrieve required information. Students will be evaluated based on logic building for the selected problem, rectification of syntax errors and expected output.

b) Mini Project Evaluation: 10 Marks

A group of 3-4 students should be assigned a real life problem statement. Students are expected to research and collect required resources to create a frontend and backend for the selected problem. Students should prepare a presentation/problem solving of 10-15 minutes. This project should be graded for 10 marks depending on the parameters as analysis, design,

conversion to relational schema, database creation and implementation for selected problem statement.

c) Regularity & Active participation: 05 Marks

END SEMESTER EXAMINATION (25 MARKS)

a) Task Execution: 10 Marks
Students will be given task (different task for every student) and will be evaluated as per the parameters mentioned in continuous assessment.
b) Results and discussion, Inferences drawn from the above task: 05 Marks
c) Oral based on entire syllabus :10 Marks

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
SBL	ITSBL301	PYTHON LABORATORY	02

Examination Scheme		
Continuous Assessment	**End Semester Exam (ESE)**	**Total**
50	50	100

Pre-requisite :

1. ESL103- Programming Laboratory-I (C)
2. ESL205- Programming Laboratory-II (Java)

Program Outcomes addressed :

1. PO 1: Engineering knowledge
2. PO 2: Problem analysis
3. PO 3: Design/Development of Solutions
4. PO 5: Engineering tool usage
5. PO 8: Individual and Collaborative team work
6. PO 9: Communication

Course Objectives :

1. To provide basics of python programming including data types, operator, conditional statements, looping statements, input, and output functions in Python
2. To impart knowledge of advanced data types of python programming.
3. To inculcate knowledge of object oriented programming concepts in python.
4. To impart with concepts of modules, packages, multithreading and exception handling.
5. To acquaint learner with File handling, GUI & database programming.
6. To impart knowledge of data visualization using Matplotlib, Data analysis using Pandas and Web programming using Flask/Django.

Module	Detailed Contents	Hrs
00.	**Course Introduction** This course provides programming concepts of python from basics to advanced concepts. Design and write fully functional Python programs using commonly used data structures, custom functions, and reading and writing to files overview of oops concepts. provides basics of creating modules, packages and importing packages and provides fundamentals of GUI designing, database connectivity and web programming, data visualization and analysis. These concepts are essential for developing skill sets to work in the field of data science.	**01**
01.	**Basics of python and introduction to advanced data types, strings and special functions .**	**11**

	Learning Objective/s: *Learner is expected to infer and apply regular programming constructs to write python programs.* *Learner is expected to illustrate and use advanced data types in python for defining an ordered and unordered collections.* *Learner is expected to summarize string functions and special functions for code optimization.*	
	Contents: Basics of Python: Variables, operators, control structures, conditional structures, functions, lists, tuples, set, dictionary and its built-in functions strings and its built in functions, map reduce and filter function. **Task 1:** Write a program to input details such as rno,name,marks in respective subject of sem I for five students , find rank of each student and display details of each student. **Task 2:** Write a program to modularize task1 using functions to find and display rank of a student. **Task 3:** Identify and use advanced data type to store students detailed information and perform built in operations and also verify the output for any 2 advanced data types identified. **Task4:** Use Map function to find grade of a student for set of iterables from task 3 and filter function to list all students whose grade is "A". **Task 5:** Use String function to search students by roll number and display grade of a particular student.	
	Self-Learning Topics: *Decorators, iterators, generator functions* ***Learning Outcomes :*** *A learner will be able to* *LO 1.1: Use variables, operators, control and conditional structures to write program in python. (P.I.-1.3.1)* *LO 1.2: Use functions in python to modularize a given program with set of statements. (P.I.-1.4.1)* *LO 1.3: Identify, use and analyze advanced data types and their built in operations that applies to a given problem. (P.I.-2.1.2)* *LO 1.4: Use dictionary to implement key value pair elements for storing data of any type (P.I.-2.2.3)* *LO 1.5: Implement string built in functions to meet functional requirements to write a program in python. (P.I.-3.2.2)* *LO 1.6: Apply Map and reduce functions to solve a given program in python as a design alternative for for each function and for loop. (P.I.-3.2.1)*	
02.	**Object oriented programming concepts in python** ***Learning Objective/s:*** *Learner is expected to summarize and apply basic concepts of object-oriented programming in python to make code more reusable and easier to work with larger programs.*	**12**
	Contents:	
	Overview of Object -oriented programming, creating classes and objects, self-Variable, constructors, operator overloading, method overloading	

	Inheritance: Types of Inheritance (Single, Multiple, Multi -level, Hierarchical) constructors in inheritance, method overriding, abstract class, abstract method, interfaces in python.,	
	Task 6: Create a class by name Teacher and add variables to store details of the teacher ,add constructor to initialize member variables and display details of teacher using methods**.** **Task 7:** Create a child class and inherit properties from the Teacher class and add methods to the child class to display details of teacher teaching subjects to sem III and V**.** **Task 8:** Use the child class created in task 7 inherit properties from the Teacher class and demonstrate method overloading in the child class. **Task 9:** Create an fees_details interface add abstract methods to it and implement methods in another class.	
	Self-Learning Topics: *Single Responsibility Principle (SRP), Open/Closed Principle (OCP).* ***Learning Outcomes:*** *A learner will be able to* *LO 2.1: Create a class and declare member variables, add constructors to initialize variables (P.I.-1.3.1)* *LO 2.2: Use concepts of inheritance to achieve code reusability for solving a problem of code maintainability. (P.I -1.4.1).* *LO 2.3: Apply method overloading to implement and integrate the modules. (P.I - 3.2.2)* *LO 2.4: Identify and use alternative for abstraction using interfaces. (P.I - 3.2.1)*	
03.	**Introduction to modules, packages, exception handling multithreading in python.** ***Learning Objective/s:*** *Learner is expected to paraphrase and implement the concepts of user defined and built in modules and packages that are useful for code modularization, code reusability.* *Learner is expected to infer and use multi-threading and exception handling in python and write own exceptions.*	**12**
	Contents: Modules: Writing modules, importing objects from modules, python built -in modules. SciPy: Linear algebra functions using Numpy and Scipy Packages: creating user defined packages and importing packages. Multi -threading: creating threads in python, exception handling: Compile time errors, Runtime errors, exceptions, types of exception, try statement, except block, raise statement, assert statement, user -defined exceptions. **Task 10:** Write a program to create a module by student details,teacher details and add the code of student details and grade calculation from task2, teacher details from task 6 .create a package and add these modules in this user defined package and execute it. **Task 11:** Create two threads by name display students details and calculate grade and add exception handling to handle name error and type errors.	

	Self-Learning Topics: *Study of media related modules in python.*	
	Learning Outcomes : *A learner will be able to* *LO 3.1: Identify and write user defined exception for a given problem. (P.I -2.1.2).* *LO 3.2: Identify and use functions of NumPy and SciPy module to perform matrix operations and solve algebraic equations. (P.I -2.2.3)* *LO 3.3: Create your own module and implement it in another file. (P.I -3.2.1)* *LO3.4: Implement user defined packages to achieve code modularity in programming. (P.I-3.4.2).*	
04.	**File handling and designing graphical user interface in python** ***Learning Objective/s:*** *Learner is expected to summarize and implement basics of File handling in Python.* *Learner is expected to summarize and create backend databases and connect GUI with databases to perform CRUD operations.*	**10**
	Contents: File Handling operations in python,Graphical user interface (GUI): different GUI tools in python (Tkinter, PyQt) Working with containers, Canvas, Frame, Widgets(Button, Label, Text, Scrollbar, Check button, Radio button, Entry, Spinbox, Message etc.) Connecting GUI with databases to perform CRUD operations. (on supported databases like SQLite, MySQL, PostgreSQL) **Task 12:** Write a program to open students details file in append mode and add extra curricular details in the file. **Task 13:** Design student registration form using TKinter. **Task 14:** Create a database to store student registration details using SQLite/MYSQL and connect to frontend created in task 13 and perform CRUD operations.	
	Self-Learning Topics: *Design GUI using Kivy, MongoDB database* ***Learning Outcomes :*** *A learner will be able to* *LO 4.1: Use basics of file handling operations in python. (P.I -1.3.1)* *LO 4.2: Use TKinter to design GUI for a given problem. (P.I.-1.4.1)* *LO 4.3: Create a database using SQLite/MySQL according to the given problem statement. (P.I-3.2.1)* *LO 4.4: Perform and integrate CRUD operations in python. (P.I -3.4.2)* *LO 4.5: Identify tools to install SQLite, MySQL. (P.I -5.1.1)* *LO 4.6: Use tools to install SQLite, MySQL and perform CRUD operations. (P.I -5.1.2)*	
05.	**Data Visualization, analysis and web programming in python** ***Learning Objective/s:*** *Learner is expected to infer concepts of data visualization and analysis of data using matplotlib and pandas.*	**14**

	Learner is expected infer skills of web programming using Flask/Django framework.	
	Content: Visualization using Matplotlib: working with plots (line plot, bar graph, histogram, scatter plot, area plot, pie chart etc.), working with multiple figures. Data manipulation and analysis using Pandas: Introduction to Pandas, importing data into Python, series, data frames, indexing data frames, basic operations with data frame, filtering, combining and merging data frames, removing duplicates. Web programming using Flask, Django, **Task 15:** Write a program to plot graph to display no of students securing grade A,B,C,D from student details file . **Task 16 :** Write a program to create a simple pandas dataframe for student details data**.** **Task 17 :** Write a program to read a csv file containing student details and identify and remove rows that contain empty cells and fix wrong data with the correct values. **Task 18 :** Develop a student details web application using flask.	
	Self-Learning Topics: *API essentials in python, pygame tool* ***Learning Outcomes:*** *A learner will be able to* *LO 5.1: Identify which graph to plot using matplotlib for a given problem. (P.I -2.1.2)* *LO 5.2: Use matplotlib library to plot various graphs for analyzing data visually according to the given problem (P.I -2.4.2)* *LO 5.3: Design and demonstrate web application using flask for a given task. (P.I -3.1.1,8.1.1,8.2.1,9.1.2,9.1.3)* *LO 5.4: Identify and use various ways of cleaning data using pandas. (P.I -3.2.2)* *LO 5.5: Identify tools to install matplotlib and pandas. (P.I -5.1.1)* *LO 5.6: Use tool to install matplotlib and pandas module. (P.I -5.1.2)*	
	Total	**60**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem
2.1.2	Identifies processes/modules/algorithms of a computer-based system and parameters to solve a problem
2.2.3	Identify existing solution/methods to solve the problem, including forming justified approximations and assumptions
2.4.2	Analyze and interpret the results using contemporary tools.
3.1.1	Able to define a precise problem statement with objectives and scope.

3.2.1	Ability to explore design alternatives.
3.2.2	Ability to produce a variety of potential design solutions suited to meet functional requirements.
3.4.2	Ability to implement and integrate the modules.
5.1.1	Identify modern engineering tools, techniques and resources for engineering activities .
5.1.2	Create/adapt/modify/extend tools and techniques to solve engineering problems.
8.1.1	Recognize a variety of working and learning preferences; appreciate the value of diversity on a team
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents
9.1.3	Create flow in a document or presentation - a logical progression of ideas so that the main point is clear

Course Outcomes :

Learner will be able to

1. Apply basic concepts of structure, syntax, and semantics of the Python language and use advanced data types and functions in python. *(LO 1.1,LO 1.2, LO 1.3,LO 1.4, LO 1.5,LO 1.6)*
2. Apply the concepts of object-oriented programming as used in Python. *(LO 2.1,LO 2.2, LO 2.3,LO 2.4)*
3. Create Python applications using modules, packages, multithreading and exception handling.(*LO 3.1,LO 3.2, LO 3.3,LO 3.4)*
4. Implement File Handling and create GUI applications and perform database operations in Python.*(LO 4.1,LO 4.2, LO 4.3,LO 4.4, LO 4.5,LO 4.6)*
5. Develop cost-effective robust applications using the latest Python trends and technologies.(*LO 5.1,LO 5.2, LO 5.3,LO 5.4,LO 5.5,LO 5.6)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITSBL301.1	3	3	3	-	-	-	-	-	-	-	-
ITSBL301.2	3	3	3	-	-	-	-	-	-	-	-
ITSBL301.3	3	3	-	-	3	-	-	-	-	-	-
ITSBL301.4	3	3	-	-	3	-	-	-	-	-	-
ITSBL301.5	3	3	3	-	3	-	-	-	3	3	-
Average	**3**	**3**	**3**	**-**	**3**	**-**	**-**	**-**	**3**	**3**	**-**

Text Books:

1. Introduction to computing and problem-solving using python, E Balagurusamy, First Edition,2017, McGraw Hill Publication.
2. Python Programming Using Problem Solving Approach, Reema Thareja, Second Edition, 2023, Oxford University. Press.
3. Python: The Complete Reference, Martin C. Brown, Fourth Edition, 2018, McGraw-Hill Publication.

Reference Books:

1. Headfirst Python- A Brain Friendly Guide, Paul Barry, 2nd Edition,2016, O'Reilly Media, Inc.
2. Problem Solving and Python Programming, Arockia Mary P, Kindle Edition,2021, Shanlax Publications.

Other Resources :

1. NPTEL Course: The Joy of Computing using Python, By Prof. Sudarshan Iyengar, Department of Computer Science and Engineering, IIT Ropar :-Web link- https://nptel.ac.in/courses/106106182/
2. NPTEL Course: Python for Data science By Prof. Ragunathan Rengasamy, Department of Computer Science and Engineering, IIT Madras :-Web link- https://archive.nptel.ac.in/courses/106/106/106106212/
3. Web link-https://www.w3schools.com/python/
4. Web link-https://www.geeksforgeeks.org/python-programming-language/
5. Web link-https://www.javatpoint.com/python-tutorial

A. IN-SEMESTER ASSESSMENT (50 MARKS)

1. Task Execution (30 Marks)

Students will be given minimum 10 tasks.

Students are expected to

i. Identify variables, data types methods/approach required to write the code for the given task and apply the same.
ii. Execute given task for different inputs and verify the result
iii. Import different Python libraries to solve given problem
iv. Implement basic file handling in Python
v. Create backend databases and connect GUI to perform CRUD operations
vi. Use data visualization tools
vii. Apply appropriate mechanisms to handle unexpected errors.

Students will be evaluated based on following:

i. Logic building for the given task (10 marks)
ii. Rectifying logical errors and syntax errors (06 marks)
iii. Well-structured and organized program (06marks)
iv. Verification of experiment output for different inputs (08 marks)

Refer the sample task given below.

Example:

Create a Teacher class to add details of the teachers their subject to respective class and display details using methods.

Students are expected to identify.

i. variables, data types methods/approach required to create teacher class and add methods to display details of a given teacher
ii. Execute given task for different inputs and verify the result
iii. Follow the coding standards
iv. Identify errors and rectify the errors.

Students will be evaluated based on following:

i. Logic building for the given task (10 marks)
ii. Rectifying logical errors and syntax errors (06 marks)
iii. Well-structured and organized program (06 marks)
iv. Verification of experiment output for different inputs (08 marks)

2. Regularity and active Participation (05 Marks)

3. Practical Test (15 Marks)

a) Task Execution: 10 Marks

Students will be given task (different task for every student) to execute and will be evaluated as per the parameters mentioned in continuous evaluation

Students will be evaluated based on following:

i. Logic building for the given task (04 marks)
ii. Rectifying logical errors and syntax errors (02 marks)
iii. Well-structured and organized program (02 marks)
iv. Verification of experiment output for different inputs (02 marks)

b) Oral based on covered syllabus: 05 Marks

B. END SEMESTER EXAMINATION (Practical & Oral Exam) (50 Marks)

a) Task Execution: 30 Marks
Students will be given task (different task for every student) to execute and will be evaluated as per the parameters mentioned below
i. Logic building for the given task (10 marks)
ii. Rectifying logical errors and syntax errors (06 marks)
iii. Well-structured and organized program (06 marks)
iv. Verification of experiment output for different inputs (08 marks)

b) Presentation of Results and conclusion, Inferences drawn: 05 Marks

c) Oral based on entire syllabus: 15 Marks

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
MNP	ITMNP301	MINI PROJECT- 1A	01

Program Outcomes addressed:

1. PO1 : Engineering knowledge
2. PO2 : Problem Analysis
3. PO3 : Design/Development of Solutions
4. PO4 : Conduct investigations of complex problems
5. PO5 : Engineering Tool Usage
6. PO6 : The Engineer & The World
7. PO7 : Ethics
8. PO8 : Individual & Collaborative team work
9. PO9: Communication
10. PO10: Project Management & Finance
11. PO11: Life-long learning

Course Objectives:

1. To familiarize students about available infrastructure at Department/Institute level, online resources, plagiarism, expectations from MP 1A and 1B, etc.
2. To guide students in identifying societal or research needs and formulating them into problem statements.
3. To facilitate problem-solving in group settings.
4. To apply basic engineering principles to address identified problems.
5. To foster self-learning and research skills.

Course Outcomes:

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

1. Identify problems based on societal or research needs and methodology for solving them.
2. Apply knowledge and skills to solve societal problems collaboratively.
3. Develop interpersonal skills necessary for teamwork.
4. Analyze, verify, and validate results effectively through various methodologies, including, test cases/benchmark data/theoretical/inferences/experiments/simulations, etc.
5. Evaluate the societal and environmental impacts of proposed solutions.
6. Adhere to standard engineering practices.
7. Excel in written and oral communication by technical report writing, oral presentation, and publishing results in
   - Research/white paper/article/blog writing/publication, etc.
   - Business plan for entrepreneurship product creation
   - Patent filing/copyright.
8. Gain technical competencies by participating in competitions, hackathons, etc.
9. Demonstrate lifelong learning capabilities through self-directed group projects.
10. Apply project management principles effectively.

Guidelines for the Mini Project

- At the beginning of semester-III, project guides are required to conduct around 4 hours' orientation sessions including following topics:
  - Familiarizing students about infrastructure available at Department/Institute level and how to use it.

- How to identify societal problems and formulate project problem statement.
- How to carry out literature survey.
- What is plagiarism and what care needs to be taken while writing a report.
- What is project report template and how it should be used.
- What are expectations from mini-projects 1A and 1B.

• Mini project may be carried out in one or more form of following:

Product preparations, prototype development model, fabrication of set-ups, laboratory experiment development, process modification/development, simulation, software development, integration of software (frontend-backend) and hardware, statistical data analysis, creating awareness in society/environment etc.

• Students must form groups of 3 to 4 members either from the same or from different departments.
• Groups should conduct surveys to identify needs and develop problem statements in consultation with faculty.
• An implementation plan in Gantt/PERT/CPM chart format covering weekly activities must be submitted.
• Each group must maintain a logbook to record weekly progress, to be verified by the faculty supervisor.
• Faculty input should emphasize guiding by faculty and self-learning by group members.
• Groups should propose multiple solutions, select the best one in consultation with the supervisor, and develop a working model.
• The solution to be validated with proper justification and report to be compiled in standard format of the Institute. Research papers, competition certificates may be submitted as part of annexure to the report.
• With the focus on self-learning, innovation, addressing societal/research/innovation problems and entrepreneurship quality development within the students through the Mini Projects, it is preferable that a single project of appropriate level and quality be carried out in two semesters by all the groups of the students.
• However, based on the individual students or group capability, with the mentor's recommendations, if the proposed Mini Project adhering to the qualitative aspects mentioned above, gets completed in odd semester, then that group can be allowed to work on the extension of the Mini Project with suitable improvements/modifications or a completely new project idea in even semester. This policy can be adopted on a case by case basis.

In-Semester Continuous Assessment and End-Semester Examination Guidelines

- The Head of the Departments will assign a guide to each of the mini-projects and shall form a progress monitoring committee. The guide will carry out weekly monitoring of the project's progress. The committee shall carry out in-semester project evaluation based on presentations with a minimum of two evaluations per semester.
- Assessment will be based on individual contributions, understanding, and responses to questions asked.
- **Continuous Assessment marks distribution in semester III (50 marks):**

- 10 marks for the Topic Approval Presentation in front of the progress monitoring committee
- 15 marks for the Mid-Semester Progress Presentation in front of the progress monitoring committee
- 20 marks for the Final Report & Presentation
- 05 marks for Regularity and active participation

- **Continuous Assessment marks distribution in semester IV (50 marks):**
  - 15 marks for the In-Semester Two Presentations
  - 10 marks for the Participation in Project Competitions, TPP, etc.
  - 20 marks for the Final Report & Presentation
  - 05 marks for Regularity and active participation

The review/progress monitoring committee will assess projects based on the following criteria.

Semester III:

- Theoretical solution completion, including component/system selection/design of software solution and cost analysis.
- Two reviews will occur:
  - The first review will focus on finalizing the problem statement (topic approval).
  - The second review will focus on finalizing the proposed solution.

Semester IV:

- Expected tasks include procuring components/systems, constructing a working prototype, and validating results based on prior semester work.
- Reviews will be conducted as follows:
  - The first review will assess the readiness to build a working prototype.
  - The second review will involve a poster presentation and demonstration of the working model in the last month of the semester.

In addition to the above-mentioned points, the following performance criteria shall be included during the in-semester continuous assessment:

1. Quality of survey and need identification.
2. Clarity and innovativeness in problem definition and solutions.
3. Requirement gathering feasibility study, cost-effectiveness, and societal impact of proposed solutions.
4. Completeness and full functioning of the working model.
5. Effective use of skill sets and engineering norms.
6. Verification & validation of the solutions/test cases.
7. Individual contributions to the group.

8. Clarity in written and oral communication.
9. Participation in technical paper presentations/project competitions/hackathon competitions, etc.

End-Semester Examination in Semester IV (50 marks):

1. Presentation and demonstration to internal and external examiners: 20 marks.
2. Emphasis on problem clarity, innovativeness, societal impact, functioning of the model, skill utilization, and communication clarity: 30 marks.

Course Type	Course Code	Course Name	Credits
HSS	HSS301	PRODUCT DESIGN	02

Program Outcomes addressed:

1. PO2 Problem analysis
2. PO3 Design/Development of solutions
3. PO5 Engineering tool usage
4. PO6 The engineer and the world
5. PO7 Ethics
6. PO8 Individual and collaborative team work
7. PO10 Project management & finance
8. PO11 Life-long learning

Course Objectives:

1. Understand the product design process and its user-centered principles.
2. Apply fundamental design principles to create innovative product designs.
3. Demonstrate proficiency in generating and evaluating design concepts through ideation techniques.
4. Evaluate and synthesize sustainable and user-centric design practices in product development.

Module	Details	Hrs.
00.	**Course Introduction –** The course Product Design provides a thorough understanding of the principles, methods, and methodologies used in developing unique and functional products. Whether you want to be an industrial designer, a UX/UI specialist, or a creative problem solver, this course will provide you the necessary knowledge and abilities to envision, develop, and revise products that fulfill user needs and market demands. Students will investigate many areas of product design, such as aesthetics, usability, and sustainability, using both theory and hands-on practice. Design thinking, prototyping, and research will demonstrate to students how to turn ideas into concrete things that improve user experience and address real-world challenges.	**01**
01.	**Introduction to Product Design** ***Learning Objective:*** *Understand the fundamental principles and key elements that contribute to effective product design.*	**3-5**
	Contents: Overview of product design process, Importance of user-centered design, Design thinking methodologies, Case studies of successful product designs, Introduction to design tools and software (e.g., Sketch, Adobe XD)	

	Learning Outcomes: *A learner will be able to* *LO 1.1: Apply design thinking methodologies to develop user-centered solutions.* (*P.I.- 2.1.1, 2.3.1, 3.2.1, 3.3.1*) *LO 1.2: Gain introductory experience with digital design tools.* (*P.I. – 5.1.1, 5.2.1*)	
02.	**Design Principles and Fundamentals** ***Learning Objective:*** *Understand and apply core design principles to create functional and aesthetically pleasing products.*	**5-7**
	Contents: Understanding design principles (e.g., balance, hierarchy, contrast), Human factors in design (ergonomics, anthropometrics), Material selection and properties, Basics of aesthetics and styling, Hands-on exercises in sketching and prototyping	
	Learning Outcomes: *A learner will be able to* *LO 2.1: Apply fundamental design principles such as balance, contrast, proportion, and harmony to create aesthetically and functionally effective designs.* (*P.I. – 3.1.5, 3.2.3, 6.1.1*) *LO 2.2: Analyze and evaluate design elements to develop user-centered solutions that enhance usability, accessibility, and overall user experience.* (*P.I. – 3.1.1, 6.2.1*)	
03.	**Concept Generation and Ideation** ***Learning Objective:*** *Develop creative ideas and transform them into viable product concepts through structured ideation techniques.*	**5-7**
	Contents: Techniques for brainstorming and idea generation, Sketching and visualization techniques, Developing design briefs and specifications, Evaluating and selecting design concepts, Design for X, Rapid prototyping methods (e.g., 3D printing, CNC machining)	
	Learning Outcomes: *A learner will be able to* *LO 3.1: Apply various ideation techniques such as brainstorming, mind mapping, and SCAMPER to generate innovative and creative product concepts*. (*P.I. – 2.4.4, 3.1.6, 3.2.1*) *LO 3.2: Develop and evaluate multiple design concepts based on user needs, feasibility, and functionality to refine ideas into viable solutions.* (*P.I. – 2.2.4, 3.1.1, 3.1.6*)	
04.	**Product Lifecycle** ***Learning Objectives:*** *Understand the stages of a product's lifecycle and how they influence design, development, and sustainability.*	**3-5**
	Contents: Detailed overview of the product development lifecycle, Cost estimation and budgeting, Marketing and Market research, Regulatory and compliance requirements (e.g., safety standards)	

	Learning Outcomes: *A learner will be able to* *LO 4.1: Understand and analyze the stages of the product lifecycle and their impact on design, marketing, and sustainability decisions.* (*P.I. – 3.1.1, 3.1.6, 6.3.2, 11.3.2*) *LO 4.2: Analyze the influence of lifecycle considerations such as material selection, manufacturing processes, and end-of-life disposal to develop sustainable and cost-effective product solutions.* (*P.I. – 3.1.5, 6.3.1, 6.4.1, 6.4.2, 11.1.1, 11.2.2*)	
05.	**User Experience (UX) Design** ***Learning Objective:*** *Design intuitive and user-friendly products by applying UX principles and usability testing.*	**3-5**
	Contents: Understanding user needs and behaviour, Usability testing and feedback gathering, Wire-framing and prototyping for digital products, Iterative design process, Accessibility and inclusive design principles	
	Learning Outcomes : *A learner will be able to* *LO 5.1: Apply UX design principles such as usability, accessibility, and interaction design to create intuitive and user-friendly products.* (*P.I. – 3.1.6, 3.3.1, 5.2.2*) *LO 5.2: Conduct user research and usability testing to analyze user needs, gather feedback, and refine designs for an enhanced user experience.* (*P.I. – 3.1.1, 3.1.6, 5.1.2, 5.2.1, 10.3.1, 10.3.2*)	
06.	**Sustainability in Product Design** ***Learning Objective:*** *Incorporate sustainable practices and materials to create environmentally responsible product designs.*	**3-5**
	Contents: Environmental impact assessment in product design, Sustainable materials and manufacturing processes, Design for disassembly and recycling, Circular economy principles Case studies of eco-friendly product designs.	
	Learning Outcomes: *A learner will be able to* *LO 6.1: Apply sustainable design principles by selecting eco-friendly materials, optimizing manufacturing processes, and minimizing environmental impact throughout the product lifecycle.* (*P.I. – 3.1.5, 6.3.2, 7.1.1, 11.3.1*) *LO 6.2: Assess the lifecycle impact of products in terms of resource consumption, carbon footprint, and end-of-life disposal to develop eco-friendly and socially responsible design solutions.* (*P.I. – 3.4.1, 6.4.1, 7.2.2, 11.3.2*) *LO 6.3: Demonstrate good communication and collaboration with interdisciplinary teams by incorporating sustainable design concepts, explaining environmental and social implications, and enabling cross-disciplinary discussions to create new, eco-friendly product solutions.* (*P.I. – 8.2.1, 8.3.1*)	
	Course Conclusion	**01**
	Total	**30**

Performance Indicators:

P.I. No.	P.I. Statement
2.1.1	Articulate problem statements and identify objectives.
2.2.4	Compare and contrast alternative solution processes to select the best process.
2.3.1	Combine scientific principles and engineering concepts to formulate model/s (mathematical or otherwise) of a system or process that is appropriate in terms of applicability and required accuracy.
2.4.4	Extract desired understanding and conclusions consistent with objectives and limitations of the analysis.
3.1.1	Recognize that need analysis is key to good problem definition.
3.1.5	Explore and synthesize engineering requirements considering health, safety risks, environmental, cultural and societal issues.
3.1.6	Determine design objectives, functional requirements and arrive at specifications.
3.2.1	Apply formal idea generation tools to develop multiple engineering design solutions.
3.3.1	Apply formal decision-making tools to select optimal engineering design solutions for further development.
3.2.3	Identify suitable criteria for the evaluation of alternate design solutions.
3.4.1	Refine a conceptual design into a detailed design within the existing constraints (of the resources)
5.1.1	Identify modern engineering tools such as computer-aided drafting, 46odelling and analysis; techniques and resources for engineering activities.
5.1.2	Create/adapt/modify/extend tools and techniques to solve engineering problems.
5.2.1	Identify the strengths and limitations of tools for (i) acquiring information, (ii) 46odelling and simulating, (iii) monitoring system performance, and (iv) creating engineering designs.
5.2.2	Demonstrate proficiency in using discipline-specific tools.
6.1.1	Identify and describe various engineering roles; particularly as pertains to protection of the public and public interest at the global, regional and local level.
6.2.1	Interpret legislation, regulations, codes, and standards relevant to your discipline and explain its contribution to the protection of the public.
6.3.1	Identify risks/impacts in the life-cycle of an engineering product or activity.
6.3.2	Understand the relationship between the technical, socio-economic and environmental dimensions of sustainability.
6.4.1	Describe management techniques for sustainable development.
6.4.2	Apply principles of preventive engineering and sustainable development to an engineering activity or product relevant to the discipline.
7.1.1	Identify situations of unethical professional conduct and propose ethical alternatives.
7.2.2	Examine and apply moral & ethical principles to known case studies.
8.2.1	Demonstrate effective communication, problem-solving, conflict resolution and leadership skills.
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts.
10.3.1	Identify the tasks required to complete an engineering activity, and the resources required to complete the tasks.
10.3.2	Use project management tools to schedule an engineering project, so it is completed on time and on budget.
11.1.1	Describe the rationale for the requirement for continuing professional development.
11.2.2	Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field.
11.3.1	Source and comprehend technical literature and other credible sources of information.
11.3.2	Analyze sourced technical and popular information for feasibility, viability, sustainability, etc.

Course Outcomes: A learner will be able to –

1. Apply design thinking methodologies effectively to solve design problems. (*LO 1.1, LO 2.1, LO 2.2, LO 3.1, LO 3.2*)
2. Demonstrate proficiency in utilizing design tools and techniques for product development. (*LO 1.2, LO 5.1*)
3. Communicate and collaborate effectively for interdisciplinary teamwork. (*LO 6.3*)
4. Create functional and aesthetically pleasing product designs. (*LO 5.2*)
5. Integrate sustainable and user-centric design principles into product development processes. (*LO 4.1, LO 4.2, LO 6.1, LO 6.2*)

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
HSS301.1	--	3	3	--	--	3	--	--	--	--	--
HSS301.2	--	--	3	--	3	--	--	--	--	--	--
HSS301.3	--	--	--	--	--	--	--	3	--	--	--
HSS301.4	--	--	3	--	3	--	--	--	--	3	--
HSS301.5	--	--	3	--	--	3	3	--	--	--	3
Average	--	**3**	**3**	--	**3**	**3**	**3**	**3**	--	**3**	**3**

Text Books :

1. "Product Design and Development" by Karl T. Ulrich and Steven D. Eppinger, published by McGraw-Hill Education; 7th edition (January 25, 2021).
2. "Engineering Design: A Project-Based Introduction" by Clive L. Dym and Patrick Little, published by Wiley; 4th edition (August 26, 2015).
3. "Universal Principles of Design" by William Lidwell, Kritina Holden, and Jill Butler, published by Rockport Publishers; Revised and updated edition (January 1, 2010).

Reference Books :

1. "Sketching: Drawing Techniques for Product Designers" by Koos Eissen and Roselien Steur, published by BIS Publishers; 2nd edition (March 1, 2011).
2. "Materials and Design: The Art and Science of Material Selection in Product Design" by Mike Ashby and Kara Johnson, published by Butterworth-Heinemann; 3rd edition (October 10, 2014).
3. "The Design of Everyday Things" by Don Norman, published by Basic Books; Revisedand expanded edition (November 5, 2013).

Other Resources :

1. NPTEL Course: Product Design and Development, Prof. Inderdeep Singh, IITRoorkee Weblink: https://onlinecourses.nptel.ac.in/noc21_me83/preview
2. NPTEL Course: Product Design and Innovation, By Prof. Supradip Das, Prof. Swati Pal, Prof. Debayan Dhar, IIT Guwahati, IIT Guwahati, Web link: https://onlinecourses.nptel.ac.in/noc21 de01/preview

Continuous Assessment – Theory - (50 Marks)

Suggested breakup of distribution

Multiple Choice Questions	10 Marks
Case Study	20 Marks
Group Project	15 Marks
Regularity and Active participation	05 Marks

Course Type	Course Code	Course Name	Credits
PCC	ITPCC405	ENGINEERING MATHEMATICS-IV	03+01*

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Exam (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20 + 25*	30	50	1.5	2	125

Pre-requisite :

Program Outcomes addressed :

1. PO1: Engineering knowledge
2. PO2: Problem analysis

Course Objectives :

1. To provide the basic knowledge on the concepts of Mathematics in the field of Engineering.
2. To build a foundation to the methodology necessary for solving problems by applying the knowledge of Mathematics to the field of Engineering.

Module	Details	Hrs
00.	**Course Introduction** Engineering Mathematics IV aims to equip students with the foundational knowledge and skills necessary for analyzing uncertainty and making informed decisions in engineering contexts. This course has many applications in Information Technology For example 1) Application of in Probability in Machine Learning, NLP part of speech Hidden Markov Model. 2) Application of in Correlation and Regression in Deep learning. 3) Application of Statistics in hypothetical or inferential learning. 4) Application of Algebraic Structure in Cyber Security and Cryptography.	02
01.	**Probability Theory and Random Variable**	06-08
	Learning Objective/s:	
	The learner will be able to analyze random variables using the basic theory of probability and will be able to apply various mathematical techniques in determining probability functions.	
	Contents:	
	Conditional Probability , Bayes Theorem, Total Probability Theorem, Definition of Random Variable. Types of Random Variable: Discrete and Continuous, Probability Mass and Density Function.	

	Self-Learning Topics:	
	Cumulative Distribution and Density Function	
	Learning Outcomes :	
	A learner will be able to *LO 1.1 Identify independents sets and disjoint sets and use its knowledge in the context of conditional probability. (P.I.-2.1.3)* *LO 1.2 Apply mathematical techniques of union, intersection and addition of sets, numbers for finding probabilities of events using Bayes' Theorem and Total Probability Theorem. (P.I.-1.1.2)* *LO 1.3 Identify if a given Random variable is Discrete or continuous in nature using existing definitions and formulas from Probability. (P.I.-2.1.1)* *LO 1.4 Apply mathematical techniques of integration and summation for finding Expectation, Variance, of the Random Variable. (P.I.-1.1.1)*	
02.	**Probability Distribution**	**06-08**
	Learning Objective/s:	
	Learner will be able to analyse and identify standard probability distribution functions and apply the knowledge of distribution for finding probabilities of various events.	
	Contents:	
	Measures of Central Tendency and Dispersion, Binomial distribution, Poisson Distribution , Fitting of a Poisson Curve, Gaussian Distribution, Normal Distribution (Standard Normal distribution , Reverse problem of Normal distribution)	
	Self-Learning Topics:	
	Joint Probability Distribution	
	Learning Outcomes :	
	A learner will be able to *LO 2.1 Apply mathematical techniques of exponents, algebra and basic probability for finding the probabilities of various events using Binomial, Poisson and Normal Distribution. (P.I.-1.1.1)* *LO 2.2 Apply the advance mathematical techniques of probability to find the probabilities the standard probability distribution (P.I. -1.1.2)* *LO 2.3 Identify the area under a Standard Normal Curve (bounded or unbounded) and use its knowledge in the context of Normal Distribution. (P.I.-2.1.3)* *LO 2.4 Identify whether Poisson distribution or Normal Distribution is applicable to a given problem using basic definitions of distribution and the data inferred from the problem. (P.I.-2.2.3)*	
03.	**Sampling Theory-I**	**05-07**
	Learning Objective/s:	
	Learner will be able to formulate the null hypothesis and apply parametric testing to test the hypothesis.	
	Contents:	
	Introduction to Sampling Theory, Testing of Hypothesis, level of significance, Critical region, One tailed and Two tailed test, Students' t-distribution. Test significance of large samples test: single mean, difference between the two means,	

	Self-Learning Topics:	
	sampling distribution of proportions	
	A learner will be able to	
	LO 3.1 Identify and test the hypothesis of significance difference between the parameter and the statistics (2.1.1) *LO 3.2 Arrive at the conclusion by testing the hypothesis of significance difference between the two means (2.4.4)* *LO 3.3 Identify and apply appropriate test to be used to test the given hypothesis.(2.1.3)* *LO 3.4 Determine the test statistics using the appropriate formula (1.1.1)* *LO 3.5 Determine frequencies fitting a particular probability distribution (1.1.2)*	
04.	**Sampling Theory-II**	**05-07**
	Learning Objective/s:	
	Learner will be able to formulate the hypothesis and apply non-parametric testing to test the it.	
	Contents:	
	Chi-square test:Test of goodness of fit , Independence of attributes (Contingency table), distribution of sample variance F-test , significant difference between variances of two Samples.	
	Self-Learning Topics:	
	Yate's Correction, ANOVA	
	Learning Outcomes :	
	A learner will be able to *LO 4.1 Identify and test the hypothesis of independence of attributes (2.1.1)* *LO 4.2 Identify and test the hypothesis of significance difference between the two variance (2.2.3)* *LO 4.3 Determine the expected frequencies of the assumption. (1.1.1)*	
05.	**Correlation and Regression**	**07-09**
	Learning Objective/s:	
	Learner will be able to analyze the mathematical dataset given and apply techniques of correlation and regression to identify the relationships between variables from the dataset.	
	Contents:	
	Correlation, Karl Pearson's coefficients of correlation(r), Spearman's Rank correlation coefficient (R): Repeated Rank, Non-repeated rank, Regression, Line of regression, Curve fitting: Linear and Second-Degree Curves.	
	Self-Learning Topics:	
	Fitting of an exponential Curve	
	Learning Outcomes :	
	A learner will be able to *LO 5.1 Identify whether Karl Pearson's or Spearman's coefficient of correlation is to be used in establishing relationship between two variables depending on the dataset given. (P.I.- 2.1.3)*	

	LO 5.2 Apply basic mathematical techniques from algebra in finding the lines of regression and regression coefficients. (P.I.-1.1.1) *LO 5.3 Apply Least Square Method to fit a particular to the given data (P.I.-1.1.2)* *LO 5.4 Identify whether a linear degree curve or a quadratic degree curve is to be fit for the given data set based on the knowledge of Curve Fitting (P.I.-2.2.3)*	
06.	**Algebraic Structure**	**07-09**
	Learning Objective/s: *The learner will be able analyze the Algebraic Structure using the basic properties.*	
	Contents:	
	Rings ,Integral domain, Fields, Ring Homomorphism Ring Isomorphism	
	Self-Learning Topics: *Orthonormal basis, Basis and Dimension.*	
	Learning Outcomes : *A learner will be able to* *LO 6.1 Apply mathematical operations defined on algebraic structures like Rings, Integral domain and Field and demonstrating closure properties under the operations. (P.I.-1.1.1)* *LO 6.2 Identify substructures within algebraic systems and the concept of homomorphism between them. (P.I.-2.1.3)* *LO 6.3 Apply the properties of homomorphism and one-one prove that the homomorphism is an isomorphism (P.I. 1.1.1)* *LO 6.4 Identify and characterize various algebraic structures based on their properties.(P.I.-2.4.1)*	
	Course Conclusion	**01**
	Engineering Mathematics plays an important role in providing the analytical tools necessary for designing, analyzing, and optimizing various electronic systems and communication networks.	
Total		**45**

Performance Indicators:

P.I. No.	**P.I. Statement**
1.1.1	Apply the knowledge of discrete structures, linear algebra, statistics and numerical techniques to solve problems.
1.1.2	Apply the concepts of probability, statistics and queuing theory in modelling of computer-based system, data and network protocols.
2.1.1	Evaluate problem statements and identifies objectives.
2.1.3	Identify mathematical algorithmic knowledge that applies to a given problem.
2.2.3	Identify existing solution/methods to solve the problem, including forming justified approximations and assumptions.
2.4.1	Applies engineering mathematics to implement the solution.
2.4.4	Arrive at conclusions with respect to the objectives.

Course Outcomes :

1. Analyse random variables and apply the concepts of probability for getting the spread of data.*(LO 1.1, LO 1.2, LO 1.3 , LO 1.4)*
2. Analyse the mathematical problem given and apply the concepts of distribution in finding probabilities. *(LO 2.1, LO 2.2, LO 2.3 , LO 2..4)*
3. Apply sampling theory principles and techniques to real-world research problems in various fields and interpret the result. *(LO 3.1, LO 3.2, LO 3.3 , LO 3.4, LO 3.5, LO 4.1, LO 4.2, LO 4.3)*
4. Analyse and interpret the data using Correlation and Regression.(*LO 5.1, LO 5.2, LO 5.3 , LO 5.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC 405.1	3	2	-	-	-	-	-	-	-	-	-
ITPCC 405.2	3	2	-	-	-	-	-	-	-	-	-
ITPCC 405.3	3	2	-	-	-	-	-	-	-	-	-
ITPCC 405.4	3	2	-	-	-	-	-	-	-	-	-
ITPCC 405.5	3	2	-	-	-	-	-	-	-	-	-
Average	3	2	-	-	-	-	-	-	-	-	-

Text Books :

1. Higher Engineering Mathematics by Grewal B. S. 38th edition, Khanna Publication 2005.
2. Advanced Engineering Mathematics by Kreyszig E. 9th edition, John Wiley.
3. A Text Book of Applied Mathematics Vol. I & II by P.N.Wartikar & J.N.Wartikar, Pune, Vidyarthi Griha Prakashan, Pune.

Reference Books :

1. Topics in Algebra by Herstein.
2. Fundamentals of Probability and Statistics V. K Kapoor and S. C. Gupta,. S . Chand, Publications
3. Advanced engineering mathematics, H.K. Das, S . Chand, Publications

Other Resources :

1. NPTEL Course: NOC: Probability for Computer Science IIT Kanpur: Prof Nitin Saxena:-Web link- https://nptel.ac.in/courses/106104233/
2. NPTEL Course: Sampling Theory, IIT Kanpur: Prof Shalabh :-Web link- https://nptel.ac.in/courses/111104073
3. NPTEL Course::- NOC: Algebra -I IMSc, Prof S Vishwanath, Prof Amritanshu Prasad Web link- https://nptel.ac.in/courses/111106137

A. IN-SEMESTER ASSESSMENT (75 MARKS)

1. Continuous Assessment - Theory (20 MARKS)

One MCQ test as per Gate exam pattern/ level: 5 Marks
One Class test: 5 Marks
One Team-pair- Solo: 5 Marks
Regularity and attentiveness: 5 Marks

Continuous Assessment - Tutorial (25 MARKS)

Minimum six Tutorials: 20 Marks

*Students must be encouraged to write at least 6 class tutorials. At least 6 Class tests will be conducted based on class tutorials on entire syllabus. Each class tests carries 20 Marks. Average will be taken of all class tests***.**

Regularity and attentiveness: 5 Marks

2. Mid Semester Exam (30 Marks)

Mid semester examination will be based on 40% to 50% of the syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20%-30% weightage, and the syllabus covered from MSE to ESE carrying 70%-80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	ITPCC406	COMPUTER NETWORK	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Exam (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite:

1. ESC203: Basic Electronics Engineering

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions
4. PO6: The Engineer and The World
5. PO7: Ethics

Course Objectives:

1. Build an understanding about the concepts and fundamentals of computer networks.
2. Perceive comprehensive knowledge about the principles, protocols, reference models (OSI and TCP/IP) and its functionalities
3. Apply the concepts of Error detection and error correction to identify the errors in data
4. Implement various routing algorithms and analyze them.

Module	Detailed Contents	Hrs
00.	**Course Introduction:** Computer Network is the foundation course which deals with concept of computer network, basis of communication, effective utilization of resources, optimize convenience and flexibility, safety standards and use of computer network in modern society.	01
01.	**Introduction to Computer Networks** ***Learning Objective:*** *To perceive the knowledge of basic network theory, functionalities of each layer of the models and to apply knowledge in designing of network.*	03-04

	Contents: Concept of computer network, categories of networks, Network Component: Hardware and software, Network topology and its types (Ring, mesh, star, bus, hybrid), Network structure and architecture (layering principles, services, protocols and standards), Network devices: Router, Hub, Switch, Repeater, Gateway, Reference Models: ISO/OSI Model and TCP/IP Model	
	Self-Learning Topic: *Identify the different devices used in network connection in college campus.* ***Learning Outcomes:*** *A learner will be able to* *LO 1.1: Apply the fundamental concepts of network theory to identify different networking devices and layered communication architectures and use it ethically to design a network as per the society's requirement. (P.I.-1.3.1,7.2.2)* *LO 1.2: Apply layering principles, services, protocols and standards in different reference models and identify alternatives to be used while designing the network as per organizations need in ethical manner. (P.I.-1.4.1, 7.1.1)* *LO 1.3: Identify different types of network topology for the design of network as per the requirement. (P.I.-2.2.4)* *LO 1.4: Analyze the results of applied topology structure and network devices as per the user's requirement. (P.I.-2.4.4)*	
02.	**Physical Layer** ***Learning Objective:*** *To know and differentiate between various transmission media characteristics and implement understanding in real time applications.* **Contents:** Guided Transmission Media: Twisted pair (STP and UTP), Coaxial (Baseband and broadband), Fibre optics, Unguided Media: Radio waves, Microwaves Infrared, Transmission Impairments, switching: Circuit-Switched Networks, Packet switching, Structure of a switch.	**03-05**
	Self-Learning Topics: *Compare and contrast various transmission media.* ***Learning Outcomes:*** *The learner will be able to* *LO 2.1: Categorize the various types of transmission media and identify the impact of transmission media on environment and its sustainability (P.I.-1.3.1, 7.1.2,7.2.2)* *LO 2.2: Illustrate structure of switch and its different types. (P.I.-1.4.1)* *LO 2.3: Determine physical connections set up to the network. (P.I.-3.1.4)* *LO 2.4: Demonstrate an ability to select optimal design scheme for user's requirement. (P.I.-3.3.2)*	

03.	**Data Link Layer** ***Learning Objectives:*** *To impart the knowledge of error detection, error correction and deploy facts to identify the errors in data. Also expected to implement the various routing algorithms.*	**08-10**
	Contents: Services, Framing, Error Control, Flow Control, Error Detection and Correction (Hamming Code, Parity, CRC, Checksum), Problems based on CRC, Hamming distance etc. Elementary Data Link protocols: Stop and Wait, Sliding Window (Go Back N, Selective Repeat), HDLC, IEEE 802.3 Ethernet, Problems on sliding window protocols. Multiple Access Protocols: ALOHA, Slotted ALOHA, CSMA, CSMA/CD and CSMA/CA.	
	Self-Learning Topics: *Differentiate link layer in IOT network and Normal Network.* ***Learning Outcomes:*** *The learner will be able to* *LO 3.1: Illustrate error control and flow control mechanism. (P.I.-1.3.1)* *LO 3.2: Apply statistics to determine error control and flow control. (P.I.-1.1.2)* *LO 3.3: Interpret different design issues at data link layers. (P.I.-2.2.4)* *LO 3.4: Implement and analyze multiple access protocols in data link layer. (P.I. 2.4.2)* *LO 3.5: Identify IEEE.802.3 Ethernet. (P.I.-3.1.4)* *LO 3.6: Implement and analyze elementary data link protocols and algorithms at data link layer for any communication network task. (P.I.-3.3.2)*	
04.	**Network Layer** ***Learning Objectives:*** *To acquaint the knowledge of various types of IP addressing and apply it in network design. Also expected to analyse and evaluate the performance of different routing protocols.*	**08-10**
	Contents: Network Layer design issues, Communication Primitives: Unicast, Multicast, Broadcast. IPv4 Addressing (class full and classless), numerical (calculation of network id, host ids, last address, first address, address space available etc.) Sub netting, Super netting, design problems, IPv4 Protocol, Network Address Translation (NAT), IPv6, Address resolution protocol (ARP). Routing algorithms: Distance Vector Routing, Shortest Path (Dijkastra), Link state routing, Protocols: RIP, OSPF, BGP	
	Self-Learning Topics: *Study difference between IPV4 and IPV6. Network Class A, B, C, D, E and subnet mask.* ***Learning Outcomes:*** *The learner will be able to* *LO 4.1: Apply the knowledge IP addressing in network design. (P.I.-1.3.1)* *LO 4.2: Apply the knowledge sub netting and super netting to design a network stack in real time application. (P.I.-1.4.1)* *LO 4.3: Identify different design issues at network layer. (P.I.-2.2.4)* *LO 4.4: Conceptualize and design a network stack using IP addressing and sub netting / super netting schemes. (P.I.- 2.4.2)* *LO 4.5: Implement algorithms at network layer like RIP, OSPF, BGP etc. and select the optimal algorithm (P.I.-3.3.2)* *LO 4.6: Explore design alternatives for user's problem. (P.I.-3.2.1)*	
05.	**Transport Layer & Session Layer**	**07-09**

	Learning Objective: *To impart the knowledge of TCP and UDP protocols to provide the communication services directly to the application processes running on different hosts.*	
	Contents: Transport Layer: Transport Layer Services, Connectionless & Connection-oriented Protocols, Transport Layer protocols: User Datagram Protocol: UDP Services, UDP Applications, Transmission Control Protocol: TCP Services, TCP Features, Segment, A TCP Connection, Windows in TCP, Flow Control, Error Control, TCP Congestion Control, TCP Timers Session Layer: Session layer design issues, Socket Programming, Session Layer protocol - Remote Procedure Call (RPC)	
	Self-Learning Topics: *List real time example of UDP and TCP.* ***Learning Outcomes:*** *The learner will be able to* *LO 5.1:Illustrate error control and flow control mechanism. (P.I.-1.3.1)* *LO 5.2:Summarize Remote Procedure Call (RPC) in session layer. ((P.I.- 1.4.1)* *LO 5.3:Identify connectionless and connection-oriented transport layer protocols. (P.I.- 2.2.4)* *LO 5.4:Implement socket programming using TCP and UDP. (P.I.- 2.4.1)* *LO 5.5:Implement the transport layer protocols like TCP, UDP and select the suitable protocol as per requirement. (P.I.- 3.3.2)* *LO 5.6:Explore design alternatives for client server programming. (P.I.-3.2.1)*	
06.	**Presentation Layer & Application Layer** ***Learning Objectives:*** *To identify the various data compression techniques used by presentation layer and apply it to prepare data for the application layer. Also expected to introduce various application layer protocols used by organizations as per the requirement.*	**08-09**
	Contents:	
	Presentation Layer: Compression: Comparison between Lossy Compression and Lossless Compression, Huffman Coding, Speech Compression, LZW, RLE, Application Layer: Domain Name Space (DNS) Electronic Mail -SMTP, POP, FTP, Telnet, WWW Application: A case study to design a network for an organization meeting the following guidelines: Networking Devices, IP addressing, Routing Protocols to be used, Services to be used (TELNET, FTP server, Web server, File server, etc)	
	Self-Learning Topics: *Difference between HTTP and FTP Protocol.* ***Learning Outcomes:*** *The learner will be able to* *LO 6.1: Illustrate various application layer protocols and design a network as per the requirement of organization by applying engineers' knowledge in the networking field. (P.I.-1.3.1, 6.1.1)* *LO 6.2: Identify the various compression techniques used by presentation layer and identify engineer's contribution to the protection of the public at different level (local, global and international) (P.I.-1.4.1, 6.2.1))* *LO 6.3: Interpret the various data compression techniques suitable for efficient communication. (P.I.-2.2.4)* *LO 6.4: Identify network devices, topology used, layering architecture as per the organization's requirement. (P.I.-2.3.2)* *LO 6.5: Implement Huffman Coding and LZW. (P.I.- 3.3.2)* *LO 6.6: Explore application layer protocols in real time application. (P.I.-3.2.1)*	

	Course Conclusion	**01**
	Total	**45**

Performance Indicators:

P. I. Number	**P. I. Statement**
1.1.2	Apply the concepts of probability, statistics and queuing theory in modeling of computer-based system, data and network protocols.
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem.
2.2.4	Compare and contrast alternative solution/methods to select the best methods
2.3.2	Identify design constraints for required performance criteria.
2.4.1	Applies engineering mathematics to implement the solution.
2.4.2	Analyze and interpret the results using contemporary tools.
2.4.4	Arrive at conclusions with respect to the objectives.
3.1.4	Ability to choose appropriate quality attributes as defined by ISO/IEC/IEEE standard.
3.2.1	Ability to explore design alternatives.
3.3.2	Consult with domain experts and stakeholders to select candidate engineering design solution for further development.
6.1.1	Identify and describe various engineering roles; particularly as pertains to protection of the public and public interest at global, regional and local level.
6.2.1	Interpret legislation, regulations, codes, and standards relevant to your discipline and explain its contribution to the protection of the public.
7.1.1	Identify situations of unethical professional conduct and propose ethical alternatives.

Course Outcomes: Learner will be able to

1. Apply the fundamentals of basic network theory and layered communication architectures to design a network for the given application. *(LO 1.1,LO 1.2, LO 1.3,LO 1.4)*
2. Identify the different types of transmission media with real time applications. *(LO 2.1, LO 2.2, LO 2.3, LO 2.4)*
3. Implement algorithms at the appropriate layer, identify and analyze error and flow control mechanisms. *(LO 3.1, LO 3.2, LO 3.3, LO 3.4. LO 4.1, LO 4.2, LO 4.3, LO 4.4, LO 4.5, LO 4.6, LO 3.5, LO 3.6, LO 5.1, LO 5.2, LO 5.3, LO 5.4, LO 5.5, LO 5.6, LO 6.1, LO 6.2, LO 6.3, LO 6.4, LO 6.5, LO 6.6)*
4. Conceptualize and design a network stack using IP addressing and sub netting / super netting schemes. *(LO 3.1, LO 3.2, LO 3.3, LO 3.4,LO 3.5, LO 3.6, LO 4.1, LO 4.2, LO 4.3, LO 4.4.LO 4.5, LO 4.6, LO 6.1, LO 6.2, LO 6.3, LO 6.4, LO 6.5, LO 6.6)*

5. Analyze network service quality performance. *(LO 3.1, LO 3.2, LO 3.3, LO 3.4.LO 3.5, LO 3.6, LO 4.1, LO 4.2, LO 4.3, LO 4.4.LO 4.5, LO 4.6,LO 5.1, LO 5.2, LO 5.3, LO 5.4,LO 5.5, LO 5.6, LO 6.1, LO 6.2, LO 6.3, LO 6.4, LO 6.5, LO 6.6)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC406.1	3	3	-	-	-	-	3	-	-	-	-
ITPCC406.2	3	-	3	-	-	-	3	-	-	-	-
ITPCC406.3	3	3	3	-	-	-	-	-	-	-	-
ITPCC406.4	3	3	3	-	-	3	-	-	-	-	-
ITPCC406.5	3	3	3	-	-	3	-	-	-	-	-
Average	**3**	**3**	**3**	**-**	**-**	**3**	**3**	**-**	**-**	**-**	**-**

Text Books:

1. Computer Networks, Andrew S. Tanenbaum, David J. Wetherall, Fifth Edition, 2011, Prentice Hall
2. Data Communication and Networking", Behrouz A. Foruzan, Fifth Edition, 2013, Science Engineering & Math Publications

Reference Books:

1. Data and Computer Communication, W. Stallings, Tenth Edition, 2014, Pearson Education
2. TCP/IP Protocol Suite, B. A. Forouzan, Fourth Edition, 2010,Tata McGraw Hill edition,
3. Information Theory, Coding and Cryptography, Ranjan Bose, Second Edition, 2008, Tata McGraw-Hill
4. Introduction to Data Compression, Khalid Sayood ,Third Edition, 2010, Morgan Kaufman

Other Resources:

1. NPTEL Course: Computer Networks and Internet Protocol by Prof. Soumya Kanti Ghosh, Prof. Sandip Chakraborty, IIT Kharagpur
   Web Link- https://onlinecourses.nptel.ac.in/noc21_cs18/preview
2. NPTEL Course: Computer Networks by Prof. Hema A. Murthy**,** IIT Madras
   Web Link- https://nptel.ac.in/courses/106106091

A. IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

a) One MCQ Test as per GATE exam pattern / level: 05 Marks

b) One Class Test:05 Marks

c) One Think Pair Share (TPS) activity: 05 Marks

d) Regularity and active participation :05 Marks

2. Mid Semester Examination (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20% to 30% weightage, and the syllabus covered from MSE to ESE carrying 70% to 80% weightage.

Course Type	Course Code	Course Name	Credits
PCC	ITPCC407	OPERATING SYSTEM	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Exam (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite:

1. ESL103: Programming Laboratory-I (C)
2. ITPCC303: Data Structures and Analysis

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis

Course Objectives:

1. Facilitate the way students, comprehend the concepts and role of Operating System.
2. Intended to guide learners to understand Process Management and Scheduling Algorithms.
3. Aid learners to learn process coordination and deadlock handling in system.
4. Assist learners in acquiring a strong comprehension of the Memory and I/O Management policies.
5. Facilitate the way to comprehend and use OS as virtual machine and its benefits.

Module	Details	Hrs
00.	**Course Introduction** This course consists of concept of operating system, basic information of computer, Purpose of computer and its various components, Memory hierarchy. This course is used to understand the interaction between various hardware's and software applications.	**01**
01.	**Introduction of Operating System** ***Learning Objective:*** *To know and understand OS and its functions to solve real time-complex issues and maximize computational speed.*	**07-08**
	Contents: Introduction: Types of Computer System, Computer architecture, OS Structure, Operations, Services Interface, System Calls, System Structure, System Design and Implementation, OS Kernel, Types of kernels, Types of special purpose OS.	
	Self-Learning Topics: *Various types of computer and network system architectures.* ***Learning Outcomes:***	

	A learner will be able to *LO 1.1: Illustrate the functions of OS. (P.I.-1.3.1)* *LO 1.2: Correlate OS functions with software and hardware mechanism. (P.I.-1.4.1)* *LO 1.3: Compare various system calls at user and kernel level. (P.I.-2.2.5)* *LO 1.4: Identify various system calls and its constraints with OS functionalities. (P.I.-2.3.2)*	
02.	**Process Management** ***Learning Objective/s:*** *To know process management and apply process and thread scheduling.* **Contents:** Process Management: Process, Scheduling Inter-process Communication, Thread and its execution, CPU Scheduling: CPU Schedulers, scheduling Criteria Scheduling Algorithms. ***Self-Learning Topics:*** *Thread scheduling and system thread functions.* ***Learning Outcomes:*** *A learner will be able to* *LO 2.1: Illustrate various process scheduling algorithms. (P.I.-1.3.1)* *LO 2.2: Apply various process scheduling methods on given set of processes and analyze the obtained result. (P.I.-1.4.1)* *LO 2.3: Compare various system calls at user and kernel level. (P.I.-2.2.5)* *LO 2.4: Optimize process scheduling by changing various parameters of processes. (P.I.-2.3.2)*	**07-09**
03.	**Process Coordination** ***Learning Objective:*** *To know the process coordination and deadlock management policies in OS.* **Contents:** Process Synchronization: Critical Section, remainder section, Process Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Deadlocks: Characterization Methods for Handling Deadlocks, Deadlock Prevention, Avoidance, Detection, Recovery. ***Self-Learning Topics:*** *Study a real time case study for Deadlock detection and recovery.* ***Learning Outcomes :*** *A learner will be able to* *LO 3.1: Summarize various inter-process coordination algorithm. (P.I.-1.3.1)* *LO 3.2: Apply various process scheduling methods on given set of processes and analyze the obtained result. (P.I.-1.4.1)* *LO 3.3: Analyze the deadlock avoidance and prevention mechanism. (P.I.-2.2.5)* *LO 3.4: Identify the suitable method to avoid and detect deadlock. (P.I.-2.3.2)*	**10-11**

04.	**Memory Management**	**06-07**
	Learning Objective: *To primary memory management policies and analyse the various page scheduling algorithms*	
	Contents: Memory Management: Hardware Address, Binding Address, Space Dynamic Loading and Linking, Swapping, Contiguous Allocation, Segmentation, Paging Structure of the Page Table, TLB, Virtual Memory Management: Demand Paging Page Replacement Algorithms, Thrashing and multiprogramming.	
	Self-Learning Topics: *Memory management for any one Operating System, Implementation of Page Replacement Algorithms.* ***Learning Outcomes :*** *A learner will be able to* *LO 4.1: Compare and contrast page scheduling algorithms. (P.I.-1.3.1)* *LO 4.2: Apply various page scheduling methods on given set of references and analyze the obtained result. (P.I.-1.4.1)* *LO 4.3: Compare and analyse page hits and page miss proportions. (P.I.-2.2.5)* *LO 4.4: Identify the effect of thrashing on page scheduling. (P.I.-2.3.2)*	
05.	**Disk and I/O Management** ***Learning Objective:*** *To know and understand File and directory and I/O management and compare various disk scheduling policies.*	**04-06**
	Contents: File system: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; RAID levels. I/O Systems: Overview I/O Hardware Application, I/O Interface, Kernel I/O, Subsystem Transforming, 1/0 Requests to Hardware Operations Performance. System Protection: Goals, Domain Access matrix, System Security: The Security Problem, Threats Encryption, User Authentication.	
	Self-Learning Topics: *File System for Linux and Windows, Features of I/O facility for different OS.* ***Learning Outcomes :***	
	A learner will be able to *LO 5.1: Elaborate allocation methods for disk data. (P.I.-1.3.1)* *LO 5.2: Apply various disk scheduling methods on given set of cylinders and analyze the obtained result. (P.I.-1.4.1)* *LO 5.3: Compare various disk scheduling policies. (P.I.-2.2.5)*	

	LO 5.4: Identify I/O policies and its effect on system performance. (P.I.-2.3.2)	
06.	**Virtual Machines**	**03-04**
	Learning Objective: *To know and understand the demonstration of virtual machine on host and guest OS.*	
	Contents: Type 0, 1, and 2 hypervisors, Advantages of virtual machines, Trap and emulate method, Handling system calls, interrupts, and privileged instructions in VMs, Binary translation and paravirtualization, Type 2 Hypervisors, Shadow and nested page tables, JVMs and application-level hypervisors, Resource allocation in virtual machines, VM migration, Storage virtualization.	
	Self-Learning Topics: *Study and use of cloud based virtual machines.*	
	Learning Outcomes : *A learner will be able to* *LO 6.1: Illustrate various types of hypervisors. (P.I.-1.3.1)* *LO 6.2: Apply virtualization types and compare the performance parameters of VM(P.I.-1.4.1)* *LO 6.3: Compare resource allocation in virtual machines. (P.I.-2.2.5)* *LO 6.4: Identify resource allocation in parallel VM. (P.I.-2.3.2)*	
	Course Conclusion	**01**
	Total	**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem.
2.2.5	Compare and contrast alternative solution processes to select the best process.
2.3.2	Identify design constraints for required performance criteria.

Course Outcomes:

Learner will be able to

1. Compare the structure and functions of Operating System. *(LO 1.1, LO 1.2, LO 1.3 ,LO1.4)*
2. Compare the performance of Scheduling Algorithms. *(LO 2.1, LO 2.2, LO 2.3 ,LO 2.4)*
3. Interpret the process coordination and deadlock management. *(LO 3.1, LO 3.2, LO 3.3 ,LO 3.4)*
4. Analyse memory and disk resource management. *(LO 4.1, LO 4.2, LO 4.3 ,LO 4.4)*
5. Identify virtual machine usage and resource allocation. *(LO 5.1, LO 5.2, LO 5.3 ,LO5.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC407.1	3	3	3	-	-	-	-	-	-	-	-
ITPCC407.2	3	3	3	-	-	-	-	-	-	-	-
ITPCC407.3	3	3	3	-	-	-	-	-	-	-	-
ITPCC407.4	3	3	3	-	-	-	-	-	-	-	-
ITPCC407.5	3	3	3	-	-	-	-	-	-	-	-
Average	**3**	**3**	**3**	-	-	-	-	-	-	-	-

Text Books:

1. Operating System Concepts, Abraham Silberschatz, Peter B Galvin, Greg Gagne, 9th Edition, Wiley India Pvt. Ltd 2018.
2. Operating Systems: Principles and Practice, Thomas Anderson and Michael Dahlin, 2nd Edition, 2014, Recursive Books.

Reference Books:

1. Operating Systems Internals and Design Principles, William Stallings, 9th Edition, 2018, Pearson.
2. Modern Operating Systems, Andrew S. Tanenbaum, Herbert Bos, 4th Edition, 2014, Pearson.

Other Resources:

1. Digital material:
   NPTEL: Introduction to Operating Systems by Prof. Chester Rebiero, IIT Madras
   Web Link-https://nptel.ac.in/courses/106106144
2. Digital material:
   NPTEL: Operating System Fundamental by Prof.Santanu Chattopadhay,IIT Kharagpur
   Web Link-https://archieve.nptel.ac.in/courses/106/105/106105214

A. IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

a) One MCQ Test as per GATE exam pattern / level: 05 Marks

b) One Class Test: 05 Marks

c) Mind Map activity: 05 Marks

d) Regularity and active participation :05 Marks

2. Mid Semester Examination (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20% to 30% weightage, and the syllabus covered from MSE to ESE carrying 70% to 80% weightage

Course Type	Course Code	Course Name	Credits
PCC	ITPCC408	SOFTWARE ENGINEERING	03

Examination Scheme					
Distribution of Marks			Exam Duration (Hrs.)		Total Marks
In-semester Assessment		End Semester Exam (ESE)			
Continuous Assessment	Mid-Semester Exam (MSE)		MSE	ESE	
20	30	50	1.5	2	100

Pre-requisite: NIL

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Development of Solutions
4. PO4: Conduct investigations of complex problems
5. PO5: Engineering tool usage

Course Objectives :

1. To familiarize learner with the basic knowledge in software engineering.
2. To acquaint leaners to identify requirements, analyse and prepare models.
3. To familiarize learner to plan, schedule and track the progress of the projects.
4. To introduce leaners to design & develop the software solutions for the growth of society.
5. To familiarize learner to demonstrate and evaluate real time projects with respect to software engineering principles.
6. To introduce leaners to apply testing and assure quality in software solution.

Module	Detailed Contents	Hrs
00.	**Course Introduction** Software Engineering Course deals with the basic principles of Software Process, with emphasis on the Process models and their usage for different types of Software. This is foundation course which deals with fundamental concepts of Requirement Analysis, Software Estimation and Scheduling, Risk and Configuration Management, Software Testing and Maintenance. The fundamental concepts of this subject are essential for working on Software Projects in Industry.	01
01.	**Introduction to Software Engineering** ***Learning Objective/s:*** *Learner is expected to know the nature of software, software process and the different process models.*	05-06
	Contents:	

	Nature of Software, Software Engineering, Software Process, Capability Maturity Model (CMM) Generic Process Model, Prescriptive Process Models: The Waterfall Model, V-model, Incremental Process Models, Evolutionary Process Models, Concurrent Models, Agile process, Agility Principles, Extreme Programming (XP), Scrum, Kanban model	
	Self-Learning Topics: *Personal and Team Process Models.* ***Learning Outcomes :*** *A learner will be able to* *LO 1.1: Apply concepts of Process models to identify which process model best suits a particular software. (P.I.-1.3.1)* *LO 1.2: Use core principles of Capability Maturity Model to understand the importance of software in applications. (P.I.-1.4.1)* *LO 1.3: Differentiate Process models to identify which process model best suits a particular software. (P.I.-2.1.3)* *LO 1.4: Identify the level of software according to CMM model. (P.I.-2.2.3)*	
02.	**Requirement Analysis** ***Learning Objective/s:*** *Learner is expected to know and understand the concepts of requirement analysis in software engineering.*	**07-09**
	Contents: Software Requirements: Functional & non-functional – user-system requirement engineering process – feasibility studies – elicitation – validation & management – software prototyping – S/W documentation – Analysis and modeling Requirement Elicitation, Software requirement specification (SRS). Unified Modeling Language diagrams.	
	Self-Learning Topics: *Prioritizing requirements (Kano diagram) - real life application case study.* ***Learning Outcomes :*** *A learner will be able to* *LO 2.1: Apply basic concepts of feasibility studies to check if a particular software is feasible for an organization. (P.I.-1.3.1)* *LO 2.2: Use requirement elicitation techniques to know about customer requirements. (P.I.-1.4.1)* *LO 2.3: Use tools to validate and manage requirements of a software. (P.I.-2.1.2)* *LO2.4: Apply fundamentals of software prototyping and create software prototype. .(P.I.-2.2.3)* *LO 2.5: Determine requirements of software and develop SRS. (P.I.-3.1.6)* *LO 2.6: Identify suitable functional requirements for producing a variety of design solutions using UML for software. (PI-3.2.2)*	
03.	**Software Estimation and Scheduling** ***Learning Objective/s:*** *Learner is expected to know and understand the concepts of software estimation and scheduling.*	**07-09**
	Contents:	
	Management Spectrum, 3Ps (people, product and process) Process and Project metrics Software Project Estimation: LOC, FP, Empirical Estimation Models - COCOMO II Model, Specialized Estimation	

	Techniques, Object based estimation, use-case based estimation Project scheduling: Defining a Task Set for the Software Project, Timeline charts, Tracking the Schedule, Earned Value Analysis	
	Self-Learning Topics: *Typical Problems with IT cost estimates.* ***Learning Outcomes :*** *A learner will be able to* *LO 3.1:Identify Process and Project metrics for a software. (P.I.-1.3.1)* *LO 3.2:Use empirical estimation models to estimate cost of software. (P.I.-1.4.1)* *LO 3.3:Identify use cases to estimate cost and resources for a software. (P.I.-2.1.3)* *LO 3.4:Compare scheduling techniques and select best scheduling technique for a software. (P.I.-2.2.4)*	
04.	**Design Engineering** ***Learning Objective/s:*** *Learner is expected to know and understand concepts of design engineering.*	**05-06**
	Contents: Design Process & quality, Design Concepts, The design Model, Pattern-based Software Design. Architectural Design: Design Decisions, Views, Patterns, Application Architectures, Modeling Component level Design: component, Designing class based components, conducting component-level design.	
	Self-Learning Topics: *Refinement aspects, Refactoring* ***Learning Outcomes :*** *A learner will be able to* *LO 4.1: Apply the concepts of Pattern based Software design to use in a software. system(P.I.-1.3.1)* *LO 4.2: Use architectural design (P.I.-1.4.1)* *LO 4.3: Identify the architectural view that best suits a software. (P.I.-2.1.3)* *LO 4.4:Analyze the class based components to be used in component level design. (P.I.-2.2.4)*	
05.	**Software Risk, Configuration Management and Software Testing** ***Learning Objective/s:*** *Learner is expected to know and understand the concepts of software risk and configuration management.*	**09-10**

	Contents: Risk Identification, Risk Assessment, Risk Projection, RMMM Software Configuration management, SCM repositories, SCM process Testing: Software Quality, Testing: Strategic Approach, Strategic Issues- Testing: Strategies for Conventional Software. Dynamic Testing: Black Box Testing: Boundary Value Analysis, Equivalence Class Testing, State Table Based testing, Cause-Effect Graphing Based Testing, Error Guessing. White Box Testing Techniques: need, Logic Coverage Criteria, Basis Path Testing, Graph Matrices, Loop Testing, Static Testing: Formal Technical Review (FTR), Walkthrough.	
	Self-Learning Topics: *Configuration Management for Web Apps.* ***Learning Outcomes :*** *A learner will be able to* *LO 5.1: Use RMMM to identify, assess and project risks in projects. (P.I.-1.3.1)* *LO 5.2: Apply SCM Process to solve an engineering problem associated with software. (P.I.-1.4.1)* *LO 5.3: Identifies processes/algorithms to find errors using testing techniques. (P.I.-2.1.3).* *LO 5.4: Compare the results of testing method used (P.I.-2.2.4).* *LO 5.5: Apply state table based to identify state of bugs. (P.I.-3.2.1)* *LO 5.6: Determine the functionalities and validate the test cases designed. (P.I.-3.4.3)* *LO 5.7: Derive appropriate procedure/algorithm, data set and test cases to find structural errors. (P.I.-4.1.2)* *LO 5.8: Outline structural bugs and logical bugs by running the test cases to refine the process over time. (P.I.-4.3.2)*	
06.	**Software Testing Tools and Quality Assurance**	**04-05**
	Learning Objective/s: *Learner is expected to know and understand concepts of software testing and assure quality of software solution.*	
	Contents: Study of testing tools**.** Dynamic and static testing tools JIRA, Bugzilla. Software Quality Assurance Task and Plan, Metrics, Software Reliability, Software Quality Management and Assurance, McCall's quality factors and criteria, ISO9000:2000, SIX sigma. Basics of Software Maintenance and Reverse Engineering.	

	Self-Learning Topics: *Web based and mobile based Software quality assurance factors and criteria.* ***Learning Outcomes :*** *A learner will be able to* *LO 6.1: Identify quality attributes for software quality assurance (P.I.-1.3.1).* *LO 6.2: Apply Software Quality Management and Assurance for to identify possible solution (P.I.-1.4.1)* *LO 6.3: Categorize automation tools for static and dynamic testing. (P.I.-3.2.2).* *LO 6.4: Use testing tool to design test cases for various modules. (P.I.-3.4.2).* *LO 6.5: Use appropriate testing tools and techniques to collect and analyze test data. (P.I.-4.3.1)* *LO 6.6: Analyze test cases stating possible errors and limitations using automation. (P.I.-4.3.2).* *LO 6.7: Use tools and techniques to design test data and test cases (P.I.-5.1.2)* *LO 6.8: Evaluate the credibility of results from tools used with reference to the accuracy and limitations. (P.I.-5.3.2)*	
	Course Conclusion	**01**
Total		**45**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem
2.1.3	Identifies mathematical algorithmic knowledge that applies to a given problem
2.2.4	Compare and contrast alternative solution/methods to select the best methods
3.1.6	Ability to develop software requirement specifications (SRS).
3.2.1	Ability to explore design alternatives.
3.2.2	Ability to produce a variety of potential design solutions suited to meet functional requirements.
3.4.2	Ability to implement and integrate the modules.
3.4.3	Ability to verify the functionalities and validate the design.
4.1.2	Ability to choose appropriate procedure/algorithm, data set and test cases.
4.3.2	Critically analyse data for trends and correlations, stating possible errors and limitations
5.1.2	Create/adapt/modify/extend tools and techniques to solve engineering problems.
5.3.2	Verify the credibility of results from tool use with reference to the accuracy and limitations, and the assumptions inherent in their use.

Course Outcomes:

1. Illustrate use of basic knowledge in software engineering. *(LO 1.1, LO1.2, LO1.3, LO1.4)*
2. Identify requirements, analyze and prepare models. *(LO 2.1, LO 2.2, LO 2.3, LO 2 .4, LO 2.5, LO 2.6)*
3. Plan, schedule and track the progress of the projects.(*LO 3.1, LO 3.2, LO 3.3, LO 3.4)*

4. Design & develop the software solutions for the growth of society. (*LO 4.1, LO 4.2, LO 4.3, LO 4.4)*
5. To demonstrate and evaluate real time projects with respect to software engineering principles. *(LO 5.1, LO 5.2, LO 5.3, LO 5 .4, LO 5.5, LO 5.6,LO 5.7, LO5.8, LO 6.1, LO 6.2, LO 6.3, LO 6 .4, LO 6.5, LO 6.6,LO 6.7, LO6.8)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITPCC408.1	3	3	-	-	-	-	-	-	-	-	-
ITPCC408.2	3	3	3	-	-	-	-	-	-	-	-
ITPCC408.3	3	3	-	-	-	-	-	-	-	-	-
ITPCC408.4	3	3	3	-	-	-	-	-	-	-	-
ITPCC408.5	3	3	3	3	3	-	-	-	-	-	-
Average	**3**	**3**	**3**	**3**	**3**	-	-	-	-	-	-

Text Books :

1. Software Engineering: A Practitioner‘s Approach, Roger Pressman, 9th edition , 2019, McGraw-Hill Publications
2. Software Engineering , Ian Sommerville, 9th edition, 2011, Pearson Education
3. Software Engineering Fundamentals , Ali Behfrooz and Fredeick J. Hudson, 9th edition, 1997, Oxford University Press.
4. The unified modeling language user guide, Grady Booch, James Rambaugh, Ivar Jacobson, Second Edition, 2005, Pearson Education

Reference Books :

1. An integrated approach to Software Engineering, Pankaj Jalote, Third edition, 2005, Springer publication.
2. Fundamentals of Software Engineering , Rajib Mall, Fifth Edition, 2014, Prentice Hall India
3. Software Engineering, Jibitesh Mishra and Ashok Mohanty, Third edition, 2011, Pearson Education
4. Software Engineering – Concepts and Practices, Ugrasen Suman, Third edition, 2013, Cengage Learning
5. Software Engineering principles and practice, Roger Pressman, Second edition , 2004, McGraw-Hill Publications

Other Resources:

1. NPTEL Course: Software Engineering By Prof. Rushikesh Joshi, Prof. Umesh Bellur, Prof. N.L. Sarda, Department of Computer Science and Engineering, IIT Bombay :-Web link- https://nptel.ac.in/courses/106/101/106101061/
2. NPTEL Course: Software Engineering By Prof. Rajib Mall, Department of Computer Science and Engineering, IIT Kharagpur :-Web link- https://nptel.ac.in/courses/106105087

A. IN-SEMESTER ASSESSMENT (50 MARKS)

1. Continuous Assessment (20 Marks)

Suggested breakup of distribution

a) One MCQ Test as per GATE exam pattern / level: 05 Marks

b) One Class test:05 Marks

c) One Mind Map activity: 05 Marks

d) Regularity and active participation :05 Marks

2. Mid Semester Examination (30 Marks)

Mid semester examination will be based on 40% to 50% syllabus.

B. END SEMESTER EXAMINATION (50 MARKS)

End Semester Examination will be based on syllabus coverage up to the Mid Semester Examination (MSE) carrying 20% to 30% weightage, and the syllabus covered from MSE to ESE carrying 70% to 80% weightage.

Course Type	Course Code	Course Name	Credits
LBC	ITLBC403	NETWORKS LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Exam(ESE)**	**Total**
25	25	50

Pre-requisite:

1. ESL103: Programming Laboratory-I (C)
2. ESL205: Programming Laboratory-II (Java)
3. ITSBL301: Python Laboratory

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/development of solutions
4. PO5: Engineering tool usage
5. PO6: The engineer and the world
6. PO7: Ethics
7. PO10: Project management and finance

Course Objectives:

1. Build an understanding about fundamental concepts of computer network, protocols, architecture and applications.
2. Demonstrate hands-on experience of computer network simulation and modelling techniques using simulation software.
3. Implement client-server socket programming.
4. Demonstrate and interpret the traffic flow and the contents of protocol frames.
5. Design and configure a network for an organization.

Module	Detailed Contents	Hrs
00.	**Course Introduction** The objective of Networks Laboratory course is to get practical knowledge of basic networking commands and working principles of various communication protocols using simulation software. Also analyses the structure and formats of TCP/IP layer protocols using network tools such as wireshark and network simulators.	
01.	**Basic Networking Commands** ***Learning Objective:*** *To impart the knowledge of basic networking commands and execute the networking commands to get network information.* **Task 1:**	**02**

	Understanding and execution of Basic networking Commands: ifconfig ,ip, traceroute, tracepath, ping, netstat, dig, nslookup, route, host, arp, hostname ***Self-Learning Topic:*** *Learn about various types of networks (LAN, WAN, MAN)* ***Learning Outcomes :*** *A learner will be able to* *LO 1.1:Comprehend how to configure network interfaces, assign IP addresses, and manage network interfaces using commands like ifconfig and ip. (P.I.- 1.3.1)* *LO 1.2:Diagnose network connectivity problems by utilizing commands like ping to check connectivity to remote hosts, traceroute and tracepath to trace the route packets take through the network, and netstat to display network statistics and active connections. (P.I.- 1.4.1)* *LO 1.3:Enhance system administration skills required for managing and maintaining computer networks in effective manner. (P.I.- 6.1.1, 6.1.2)*	
02.	**Network equipment** ***Learning Objective:*** *To be aware about network devices and exploit it in network design as per the users/ organization's requirement.* **Task 2:** Hands-on on network equipment. • Switches, Router • Hardware Firewall ***Self-Learning Topic:*** *Learn about other networking devices like repeater, gateway, hub etc.* ***Learning Outcomes :*** *A learner will be able to* *LO 2.1: Gain practical experience in configuring switches, routers, and hardware firewalls. (P.I.- 1.3.1)* *LO 2.2: Implement firewall rules in certain scenario. (P.I.- 1.4.1)* *LO 2.3: Prioritize network traffic, manage bandwidth usage, and improve network performance for critical applications. (P.I.- 2.4.2)* *LO 2.4: Apply security best practices by configuring access control lists on routers and firewalls. (P.I.- 2.2.4)* *LO 2.5: Identify anomalies, troubleshoot performance issues, and optimize network design. (P.I.- 3.2.1)* *LO 2.6: Gain experience in using network monitoring and analysis tools. (P.I.-3.3.2)* *LO 2.7: Analyze the requirements of users or organizations and select appropriate network devices to design efficient network architectures. (P.I.- 6.1.1, 6.1.2)*	**02**
03.	**Basics of Network simulation and simulation of Network Topology with different Protocols** ***Learning Objective/s:*** *To illustrate the basics of Network simulation. Also expected to implement different routing protocols and analyze it to make the best decisions for your network communication, security and management needs.*	**08**
	Task 3: Installation and configuration of NS-2 simulator. Write TCL scripts to create topologies.	

	Task 4: Write TCL scripts for topology with graphical simulation of traffic consideration (TCP, UDP) using NAM and plot the graph. **Task 5**: Implement distance vector routing protocol in NS2.	
	Self-Learning Topic: *Implement link state routing protocols in NS2.* ***Learning Outcomes:*** *The learner will be able to* *LO 3.1: Invoke analytical studies of Computer Networks through network simulation societal impact of network technologies, including considerations related to security, privacy, and accessibility, and propose solutions that align with ethical and societal standards. (P.I.- 1.3.1, 6.1.1,6.2.1)* *LO 3.2: Demonstrate a comprehensive understanding of the fundamentals of network simulation. (P.I.- 1.4.1)* *LO 3.3: Identify modern engineering tools to simulate TCP, UDP, Link state routing protocols to emulate real-world networking scenarios. (P.I.- 5.1.1)* *LO 3.4: Identify the strengths and limitations of modern tools used for simulating and monitoring system performance. (P.I.- 5.2.1)* *LO 3.5: Develop project management skills by planning and executing network simulation projects, including defining project scopes, setting timelines, allocating resources, and evaluating project outcomes in terms of cost-effectiveness and efficiency. (P.I.- 10.1.2,10.3.1)*	
04.	**Socket Programming** ***Learning Objective:*** *To impart the knowledge about socket programming and implement it to create client and server applications in order to exchange information between processes on the same machine or across a network, allow access to centralized data.* **Task 6**: To study and Implement Socket Programming using TCP. **Task 7:** To study and Implement Socket Programming using UDP.	**04**
	Self-Learning Topics: *Learn about Echo Client and Server.* ***Learning Outcomes:*** *The learner will be able to* *LO 4.1: Summarize the underlying principles of networking protocols, communication mechanisms, and socket API functions. (P.I.- 1.3.1)* *LO 4.2: Implement client-server applications using socket programming. (P.I.- 1.4.1)* *LO 4.3: Utilize socket programming libraries and development tools to implement client-server applications and analyze the societal implications of networked systems. (P.I.- 5.1.1,6.1.1,6.1.2)* *LO 4.4: Demonstrate competence in employing modern tools for network application development. (P.I.- 5.2.1)*	
05.	**Protocol analyzer and traffic analysis** ***Learning Objective:*** *To know and illustrate various network protocol analyzer tools and use it to analyze the traffic with the help of different performance measures*. **Task 8:** Study various network protocol analyzer tools and install one of the network protocol analyzer tools. **Task 9:** Analyze the network traffic using one of the network protocol analyzer tools.	**04**

	Self-Learning Topics: *Study tcpdump, Windump, Microsoft Message Analyzer, Ettercap, ISOFT Smart Sniff protocol analyzer tools and check the performance.* ***Learning Outcomes:*** *The learner will be able to* *LO 5.1: Demonstrate the ability to measure and analyze network parameters for high throughput networks. (P.I.- 1.3.1)* *LO 5.2: Apply appropriate measurement techniques to assess network performance. (P.I.- 1.4.1)* *LO 5.3: Identify network protocol analyzer tools analyze the traffic with the help of different performance measures. (P.I.- 5.1.1)* *LO 5.4: Identify the strengths and limitations of protocol analyzer tools used for simulating and monitoring system performance. (P.I.- 5.2.1)*	
06.	**Network Design** ***Learning Objective:*** *To know and summarize IP addressing, networking protocols, layering architecture and implement it to design and configure a network for an organization.* **Task 10:** Perform remote login using Telnet Server. **Task 11:** Perform File Transfer and Access using FTP. **Task 12:** Design a network for an organization using the concepts of Addressing (IP Address Assignment), Naming (DNS) and Routing. Also mention the internetworking devices used.	**10**
	Self-Learning Topics: *Any case study: Design network for any scenario as per users requirement.*	
	Learning Outcomes: *The learner will be able to* *LO 6.1: Evaluate the societal impact of network technologies, including considerations related to security, privacy, and accessibility, and propose solutions that adhere to ethical and societal standards. (P.I.- 1.3.1,6.1.1,7.1.1)* *LO 6.2:Develop effective communication, collaboration, and teamwork skills, demonstrating the ability to contribute positively to group dynamics and achieve common goals in network engineering projects.(P.I.- 1.4.1,6.2.1,7.2.2)*	
	Total	**30**

Performance Indicators:

P. I. Number	**P. I. Statement**
1.3.1	Apply engineering fundamentals.
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem.
2.2.4	Compare and contrast alternative solution/methods to select the best methods
2.4.2	Analyse and interpret the results using contemporary tools.
3.2.1	Ability to explore design alternatives.

3.3.2	Consult with domain experts and stakeholders to select candidate engineering design solution for further development.
5.1.1	Identify modern engineering tools, techniques and resources for engineering activities
5.2.1	Identify the strengths and limitations of tools for (i) acquiring information, (ii) modelling and simulating, (iii) monitoring system performance, and (iv) creating engineering designs.
6.1.1	Identify and describe various engineering roles; particularly as pertains to protection of the public and public interest at global, regional and local level.
6.2.1	Interpret legislation, regulations, codes, and standards relevant to your discipline and explain its contribution to the protection of the public.
7.1.1	Identify situations of unethical professional conduct and propose ethical alternatives.
7.2.2	Examine and apply moral & ethical principles to known case studies.
10.1.2	Analyse different forms of financial statements to evaluate the financial status of an engineering project
10.3.1	Identify the tasks required to complete an engineering activity, and the resources required to complete the tasks

Course Outcomes:

Learner will be able to

1. Execute and evaluate network administration commands. Also use network equipment to design the network for certain scenario. *(LO 1.1, LO1.2, LO1.3)*
2. Invoke analytical studies of computer networks through network simulation. *(LO 1.1, LO1.2, LO1.3, LO 2.1, LO 2..2, LO2.3,LO 2.4 , LO 2.5 , LO 2.6, LO 2.7)*
3. Implement the socket programming for client server architecture. (*LO 3.1, LO 3..2, LO3.3,LO 3.4 , LO 3.5 ,LO 4.1, LO 4..2, LO 4.3,LO 4.4)*
4. Measure and analyse the network parameters for a high throughput network. *(LO 5.1, LO 5.2, LO 5.3,LO 5.4)*
5. Design a network using NS-3 toolkit and its importance in designing a real network. . *(LO 6.1, LO 6.2)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITLBC403.1	3	3	-	-	-	3	-	-	-	-	-
ITLBC403.2	3	3	3	-	3	-	-	-	-	-	-
ITLBC403.3	-	3	3	-	3	-	-	-	-	-	-
ITLBC403.4	3	3	-	-	3	-	-	-	-	-	-
ITLBC403.5	-	3	3	-	3	-	-	-	-	-	-
Average	**3**	**3**	**3**	**-**	**3**	**3**	**-**	**-**	**-**	**-**	**-**

Text Books:

1. Computer Networks, Andrew S. Tanenbaum, David J. Wetherall, Fifth Edition, 2011, Prentice Hall
2. Data Communication and Networking", Behrouz A. Foruzan, Fifth Edition, 2013, Science Engineering & Math Publications

Reference Books:

1. Data and Computer Communication, W. Stallings, Tenth Edition, 2014, Pearson Education
2. TCP/IP Protocol Suite, B. A. Forouzan, Fourth Edition, 2010,Tata McGraw Hill edition,
3. Information Theory, Coding and Cryptography, Ranjan Bose, Second Edition, 2008, Tata McGraw-Hill
4. Introduction to Data Compression, Khalid Sayood ,Third Edition, 2010, Morgan Kaufman

Other Resources:

1. NPTEL Course: Computer Networks and Internet Protocol by Prof. Soumya Kanti Ghosh, Prof. Sandip Chakraborty, IIT Kharagpur
   Web Link- https://onlinecourses.nptel.ac.in/noc21_cs18/preview
2. NPTEL Course: Computer Networks by Prof. Hema A. Murthy**,** IIT Madras
   Web Link- https://nptel.ac.in/courses/106106091

A. IN-SEMESTER ASSESSMENT (25 MARKS)

1. Continuous Assessment (25 Marks)

Suggested breakup of distribution

a) Task Execution :10 Marks

Students will be given 10 tasks based on list as per mentioned in the syllabus. Each task carries 10 Marks. Average will be taken of all tasks. Students are expected to execute the program for given task with network simulator, protocol analyser and java. Students will be evaluated based on logic building for the given task, expected output and analysis of received results.

b) Practical Test: 10 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

c) Regularity & Active participation: 05 Marks

B. END SEMESTER EXAMINATION (Practical and Oral Exam) (25 Marks)

Suggested breakup of distribution

a) Task Execution: 10 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

b) Results and discussion, Inferences drawn based on above task: 05 Marks
c) Oral based on entire syllabus :10 Marks

Two examiners, one Internal and one External will do the evaluation

Course Type	Course Code	Course Name	Credits
LBC	ITLBC404	LINUX LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Exam(ESE)**	**Total**
25	25	50

Pre-requisite:

1. ESL103: Programming Laboratory-I (C)

Program Outcomes addressed:

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of Solution

Course Objectives:

1. To train learners with demonstration and installation of OS guide with basic commands.
2. To instruct learners about demonstration of file and directory management services using Linux commands.
3. To instruct learners about demonstration of user, memory and process management services using Linux commands.
4. To provide hands-on for learners for demonstration of basic shell script.
5. To provide hands-on for learners for demonstration of shell script for advance applications.
6. To prepare students to execute Perl and awk script.

Module	Detailed Contents	Hrs
00.	**Course Introduction** This course discovers the significance of Linux operating System, an open-source operating system that offers security, customization, and cost-effectiveness. The Linux command line is a powerful tool that can be used to save time in a variety of ways. By using the command line, users can quickly navigate through the file system, search for files, create and delete directories, and even run programs. This course is majorly delivers the command line interaction of user with OS.	**01**
01.	**Basic Utility Commands** ***Learning Objective:*** *Demonstration of Linux Installation and Basic Commands.* **Contents:** Utility Commands: echo, clear, exit, date, time, uptime, cal, cat, man, which, history, id, pwd, whoami, pr, lp, lpr, lpstat, lpq, lprm, cancel, mail. cal: The calendar, date: Displaying the system date, echo: Displaying message, printf: An alternative to echo, bc: The calculator, script: Recording your session, Email basics, mailx: The universal mailer passwd: Changing your password, who: Who are the users?, uname: Knowing your machine characteristics, tty: Knowing your terminal, stty: Displaying and setting terminal characteristics.	**03**

	Task 1: Execute various utility commands in Linux.	
	Self-Learning Topics: *Demonstrate the commands line OS installation, memory partitions.* ***Learning Outcomes:*** *A learner will be able to* *LO 1.1: Compile and execute C program on Ubuntu OS using CLI. (P.I.-1.3.1)* *LO 1.2: Code array as 1D and 2D data structure. (P.I.-1.4.1)* *LO 1.3: Identify and use the mail commands for local smtp communication (P.I.-2.2.3)* *LO 1.4: Install Linux OS as single machine or alongside of Windows 10,11. (P.I.-2.3.1)* *LO 1.5: Demonstrate basic utility commands in Linux. (P.I.- 3.4.2)* *LO 1.6: Validate the credentials of system and other users in OS. (P.I.- 3.4.3)*	
02.	**File and Directory Management Commands** ***Learning Objective:*** *To understand and perform file and directory handling in Linux using command line interface.*	**06**
	Contents: The file: Ordinary file, Directory file, Device file, Filename, the parent-child relationship, UNIX file system tree, The Unix file system, the home directory, pwd: Checking your current directory, cd: Changing the current directory, mkdir: Making directories, rmdir: Removing directories, ls: Listing directory contents, Absolute pathnames, Relative pathnames Handling ordinary files, cat: Displaying and creating files, cp: Copying file, rm: Deleting files, mv: Renaming files, more: Paging output The lp subsystem: printing a file, file: knowing the filetypes wc: Counting lines, words and characters, od: Displaying data in octal, cmp: Comparing two files, comm: What is common, diff: Converting one file to other, gzip and gunzip: Compressing and decompressing files and folders. tar: The archival program, zip and unzip: Compressing and archiving together. Basic file attributes, ls -l: Listing file attributes, the -d option: Listing directory attributes. File ownership, File permissions, chmod: Changing file permissions, directory permission, Changing file ownership, chown: Changing file owner, chgrp: Changing group owner **Task 2:** Execute various file permission commands. **Task 3:** Execute file and directory management commands.	
	Self-Learning Topics: *Commands to automate the disk space management and user alert by OS.*	
	Learning Outcomes: *A learner will be able to* *LO 2.1: Associate file access rights to owner, group user and all system users. (P.I.-1.3.1)* *LO 2.2: Identify and change the mode of files and folders. (P.I.-1.4.1)* *LO 2.3: Apply the commands to compress the files and folders. (P.I.-2.2.3)* *LO 2.4: Identify and implement directory management (P.I.-2.3.1)* *LO 2.5: Select various types of files and count total number of files and directories. (P.I.- 3.4.2)* *LO 2.6: Identify issues in file and directory management (P.I.- 3.4.3)*	

03.	**Memory, Process and User Management Commands** ***Learning Objective:*** *To understand the tasks in memory and user management and user management and able execute relevant commands on CLI.*	**06**
	Contents: a) Execution of Memory Management Commands like free, /proc/meminfo, top, htop, df, du, vmstat, demidecode, sar, pagesize, etc. b) Execution of Process Management Commands like ps, pstree, nice, kill, pkill, killall, xkill, fg, bg, pgrep, renice, etc. c)Execution of User Management Commands like who, whoami, su, sudo, login, logout, exit, passwd, useradd/adduser, usermod, userdel, groupadd, groupmod, groupdel, gpasswd, chown, chage, chgrp, chfn, etc. **Task 4:** Execute memory management commands. **Task 5:** Create the process and kill using pid. **Task 6:** Execute user management commands.	
	Self-Learning Topics: *User and group management using shell script.*	
	Learning Outcomes: *A learner will be able to* *LO 3.1: List the process and its features. (P.I.-1.3.1)* *LO 3.2: Fetch the process id by using commands. (P.I.-1.4.1)* *LO 3.3: Execute command for closing the process. (P.I.-2.2.3)* *LO 3.4: Manage the users and their permissions. (P.I.-2.3.1)* *LO 3.5: Use of advance commands for various user and process management activities. (P.I.- 3.4.2)* *LO 3.6: Execute user and memory management from CLI Linux. (P.I.- 3.4.3)*	
04.	**Basic Shell Script** ***Learning Objective:*** *To understand Linux shell processing and syntax associated with it to design batch files.*	**04**
	Contents: Write a basic shell script: with syntax of read, write statements, loop and control statements. **Task 7:** Write shell script for following problems. i. Write a shell script to perform arithmetic operations. ii. Write a shell script to calculate simple interest. iii. Write a shell script to determine largest among three integer numbers. iv. Write a shell script to determine a given year is leap year or not. *v.* Write a shell script to print multiplication table of given number using while statement.	
	Self-Learning Topics: *Execute shell script for array, structures and composite data types.* ***Learning Outcomes:*** *A Learner will be able to* *LO 4.1: Create shell file. (P.I.-1.3.1)*	

	LO 4.2: Change the mode of execution of shell script (P.I.-1.4.1) *LO 4.3: Execute the shell script. (P.I.-2.2.3)* *LO 4.4: Identify the conditions and learn to write the conditional statements in shell script. (P.I.-2.3.1)* *LO 4.5: Implement, for and while loop statement in shell script. (P.I.- 3.4.2)* *LO 4.6: Develop the shell script to handle device and network operations. (P.I.- 3.4.3)*	
05.	**Shell script for File and User Management** ***Learning Objectives:*** *To understand Linux shell processing and syntax associated with it to design batch files which are used for user and file management operations.*	**06**
	Contents: Advance shell script: with file, directory and user management functions. **Task 8:** Write an advanced shell script for file and user management. i. Write shell script to check logged-in user details and number of logged-in user. ii. Write a shell script to read and check if the directory / file exists or not, if not make the directory / file. iii. Write a shell script using grep command to find the number of words, characters, and words and lines in a file. iv. Write shell script to calculate total number of files and folder in specified folder. v. Write a shell script that creates a directory and a file within it. Insert the contents into the file and duplicate the source folder to the specified target location. .	
	Self-Learning Topics: *Code shell script for network and infrastructure security.* ***Learning Outcomes:*** *A learner will be able to* *LO 5.1: Code shell script to create folder. (P.I.-1.3.1)* *LO 5.2: Code shell script to create file and content into it. (P.I.-1.4.1)* *LO 5.3: Manage various tty users using shell script. (P.I.-2.2.3)* *LO 5.4: Fetch user details using shell script. (P.I.-2.3.1)* *LO 5.5: Develop Shell script to process the list. (P.I.- 3.4.2)* *LO 5.6: Develop Shell script to take backup of device on another connected device. (P.I.-3.4.3)*	
06.	**Execute Perl and Awk Script** ***Learning Objective:*** *To implement complex problems using data structures and algorithms, and analysing data structures inside data management models.*	**04**
	Contents Introduction and use of Perl and Awk script, Syntax in Perl, Syntax in awk, Control and conditional statements in Perl and Awk script. Perform Perl and AWK script for **Task 9:** Study and Execute Perl script. (i) Write a Perl script to sort elements of an array. (ii) Write a Perl script to check a number is prime or not.	

	(iii) Write Perl script to read and write number array. **Task 10:** Study and Execute Perl script. (i) Write an awk script to print all even numbers in a given range. (ii) Write an awk script to develop a Fibonacci series (take user input for number of terms). (iii)Write awk script to find factorial of number.	
	Self-Learning Topics: *Perl script to develop applications in open-source systems.* ***Learning Outcomes*** *A learner will be able to* *LO 6.1: Create Perl and awk file. (P.I.-1.3.1)* *LO 6.2: Associate the path in Perl file. (P.I.-1.4.1)* *LO 6.3: Execute Perl files. (P.I.-2.2.3)* *LO 6.4: Solve various problems using Perl script. (P.I.-2.3.1)* *LO 6.5: Use Perl to develop open-source system. (P.I.- 3.4.2)* *LO 6.6: Develop the shell script to handle device and network operations. (P.I.- 3.4.3)*	
	Total	**30**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem.
2.2.3	Identify existing solution/methods to solve the problem, including forming justified approximations and assumptions.
2.3.1	Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.
3.4.2	Ability to implement and integrate the modules.
3.4.3	Ability to verify the functionalities and validate the design.

Course Outcomes:

Learner will be able to

1. Use the basic utilities commands for OS installation. *(LO 1.1,LO 1.2, LO 1.3,LO 1.4, LO 1.5,LO 1.6)*
2. Study and use file and directory management commands in Linux. *(LO 2.1,LO 2.2, LO 2.3,LO 2.4, LO 2.5, LO 2.6, LO 3.1,LO 3.2, LO 3.3,LO 3.4, LO 3.5,LO 3.6)*
3. Study and use advance commands in Linux .*(LO 4.1,LO 4.2, LO 4.3,LO 4.4, LO 4.5,LO 4.6)*
4. Execute basic and advance shell script programs.*(LO 5.1,LO 5.2, LO 5.3,LO 5.4,LO 5.5,LO 5.6)*
5. Execute Perl and awk script. *(LO 6.1,LO 6.2, LO 6.3,LO 6.4, LO 6.5,LO 6.6)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITLBC404.1	3	3	3	-	-	-	-	-	-	-	-
ITLBC404.2	3	3	3	-	-	-	-	-	-	-	-
ITLBC404.3	3	3	3	-	-	-	-	-	-	-	-
ITLBC404.4	3	3	3	-	-	-	-	-	-	-	-

ITLBC404.5	3	3	3	-	-	-	-	-	-	-	-
Average	**3**	**3**	**3**	-	-	-	-	-	-	-	-

Text Books:

1. Unix Concepts and Applications, S. Das, 4th edition., 2017, McGraw Hill.
2. Mastering Unix Shell Scripting, R. Michael, 2nd edition, 2008, Wiley.
3. Linux Experiments and Open-Source Technologies, D. Ambawade, D. Shah, 2nd edition, 2014, Dreamtech Press.

Reference Books:

1. Unix Shell Programming, Y. Kanetkar, 3rd edition, 2003, BPB Publications.
2. Unix and Shell Programming, B. Forouzan and R. Gilberg, 4th edition, Cengage Learning, 2003.

Other Resources:

1. Digital material:
   Web Link- Install Ubuntu desktop | Ubuntu
2. Web Link- The Linux Command Handbook – Learn Linux Commands for Beginners (freecodecamp.org)

A. CONTINUOUS ASSESSMENT (25 MARKS)

Suggested breakup of distribution

a) Command based script execution :10 Marks

Students will be given 10 tasks based on list as per mentioned in the syllabus. Each task carries 10 Marks. Average will be taken of all tasks. Students are expected to execute the basic utility and advanced commands of Linux operating system also they have to perform command based shell, Perl, AWK script. Students will be evaluated based on logic building for the given script and expected output.

b) Practical Test: 10 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

c) Regularity & Active participation: 05 Marks

B. END SEMESTER EXAMINATION (Practical and Oral Exam) (25 Marks)

Suggested breakup of distribution

a) Command based script **e**xecution: 10 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

b) Results and discussion, Inferences drawn: 05 Marks

c) Oral based on entire syllabus :10 Marks

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
LBC	ITLBC405	SOFTWARE DEVELOPMENT LABORATORY	01

Examination Scheme		
Continuous Assessment	**End Semester Exam(ESE)**	**Total**
25	25	50

Pre-requisite:

1. ESL103: Programming Laboratory-I (C)
2. ESL205: Programming Laboratory-II (Java)
3. ITSBL301 : Python Laboratory

Program Outcomes addressed:

1. PO 1: Engineering knowledge
2. PO 2: Problem analysis
3. PO 3: Design/Development of Solutions
4. PO 4: Conduct investigations of complex problems
5. PO 5: Engineering tool usage
6. PO 6: The Engineer and The World
7. PO 8: Individual and collaborative team work
8. PO 9: Communication
9. PO 10: Project management and finance

Course Objectives:

1. To familiarize leaners, the basics of software engineering principles.
2. To make learner understand documentation Requirements, analysis, planning and scheduling.
3. To acquaint leaners with skills to design test case plans for testing software using dynamic testing techniques.
4. To make learner able to apply knowledge of automation testing.
5. To make leaner infer and apply quality assurance and management.

Module	Detailed Contents	Hrs
00.	**Course Introduction** This course deals with basics of software engineering principles, phases of software development life cycle, various process models which will be helpful to follow the process during software development, identify early challenges or cost constraints to know team's development process for the future. Software Testing tools are used in industries for automating the entire testing process. Software QA and testing helps prevent defects from reaching end-users. By detecting issues earlier in software development cycles through extensive testing, problems can be fixed before software gets deployed.	

01.	**Introduction to concepts of software engineering principles and software development process.** ***Learning Objective/s:*** *To infer basics of software engineering principles and software development life cycle and apply it for the project development process.* *Learner is expected to gain knowledge of process models and also expected to determine the right process followed for software development.*	**08**
	Task 1: Select a problem statement and write case study on Software Development Life Cycle for the same. **Task 2:** Implement Waterfall Model and Agile model for the selected problem statement. **Task 3:** Analyze the difference between waterfall model and agile model.	
	Self-Learning Topics: *Personal and Team Process Models.* ***Learning Outcomes :*** *A learner will be able to* *LO 1.1: Use software engineering principles to identify and document phases of software development life cycle. (P.I.-1.3.1,9.1.2)* *LO 1.2: Apply software engineering principles to solve an engineering problem and demonstrate the progress of project development for the selected problem statement. (P.I.-1.4.1,9.3.2)* *LO 1.3: Illustrate design alternatives for the progress of software development. (P.I.-3.2.1)* *LO 1.4: Select optimal process model for further development. (P.I.-3.4.1)* *LO 1.5: Identify the team's development process. (P.I.-2.1.1,10.3.1)* *LO 1.6: Analyze and estimate cost incurred to select best process and also analyze limitations of process models and select the most appropriate process model based on economic and financial considerations. (P.I.-2.2.5, 10.2.1)*	
02.	**Software Requirement Analysis and scheduling** ***Learning Objective/s:*** *To paraphrase and apply knowledge of documentation requirements, analysis, planning and scheduling for software development.*	**04**
	Task 4: Write software requirement specification, Work breakdown structure (WBS). **Task 5:** Prepare Gantt chart for the selected problem statement	
	Self-Learning Topics: *prioritizing requirements, Typical Problems with IT Cost Estimates.* ***Learning Outcomes:*** *A learner will be able to* *LO 2.1: Apply basics of software engineering to prepare SRS document for the selected problem statement. (P.I.-1.3.1)* *LO 2.2: Model WBS to schedule and allocate various task during SDLC. (P.I.-1.4.1)* *LO 2.3: Identify and document system requirements for the software development process. (P.I.-3.1.2)* *LO 2.4: Illustrate Gantt chart to maintain timeline for the software development process. (P.I.-3.1.6)* *LO 2.5: Use tools and techniques to prepare timeline chart P.I.- (5.1.2)*	

	LO 2.6: Identify limitations of the tools used to derive timeline chart. (P.I.-5.3.1)	
03.	**Software Testing Techniques** ***Learning Objective/s:*** *To illustrate and apply black box testing techniques for designing test cases to identify and analyze structural errors.* *Learner is expected to infer knowledge of white box testing techniques and use white box testing techniques to ensure that all the internal components work as expected and are error-free.*	**08**
	Task 6: Design test cases using black box testing techniques for the selected project, execute the test cases and discuss the results. **Task 7:** Design test cases using path testing technique and execute test cases for the same selected problem statement to identify and fix the logical errors.	
	Self-Learning Topics: *Select the test cases (positive and negative scenarios) for the selected system and Design test cases for the system using any two studied testing techniques* ***Learning Outcomes :*** *A learner will be able to* *LO 3.1: Use BVC to test the boundary values of the input. (P.I.-1.3.1)* *LO 3.2: Apply black box and white box testing to solve an engineering problem associated to input classes. (P.I.-1.4.1)* *LO 3.3: Identify processes/algorithms to find errors using testing techniques. (P.I.-2.1.2).* *LO 3.4: Compare the results of testing method used (P.I.-2.2.4).* *LO 3.5: Apply state table based testing to identify state of bugs. (P.I.-3.2.1)* *LO 3.6: Determine the functionalities and validate the test cases designed. (P.I.-3.4.3)* *LO 3.7: Derive appropriate procedure/algorithm, data set and test cases to find structural errors. (P.I.-4.1.2)* *LO 3.8: Outline structural bugs and logical bugs by running the test cases to refine the process over time. (P.I.-4.3.2)*	
04.	**Automation and Testing Tools** ***Learning Objective/s:*** *To infer knowledge of easy approach of testing using automated tools and also expected to select and use automation tools which mainly focuses on reducing manual human activity.* *Learner is expected to summarize effectiveness, and coverage of software testing to save time and effort using automated tools. Also expected to compare the actual output against the expected outcome using testing tool.*	**06**
	Task 8: Study and categorize various testing tools and implement anyone open-source testing tool. **Task 9:** Study and implementation of anyone bug tracking tool.	
	Self-Learning Topics: *Testing tool for agile based model* ***Learning Outcomes:*** *A learner will be able to* *LO 4.1: Infer need of automation in testing. (P.I.-1.3.1).* *LO 4.2: Identify automation tools for testing. (P.I.-1.4.1)* *LO 4.3: Categorize automation tools for static and dynamic testing. (P.I.-3.2.2).*	

	LO 4.4: Use testing tool to design test cases for various modules. (P.I.-3.4.2). *LO 4.5: Use appropriate testing tools and techniques to collect and analyze test data. (P.I.-4.3.1)* *LO 4.6: Analyze test cases stating possible errors and limitations using automation. (P.I.-4.3.2).* *LO 4.7: Use tools and techniques to design test data and test cases (P.I.-5.1.2)* *LO 4.8: Evaluate the credibility of results from tools used with reference to the accuracy and limitations. (P.I.-5.3.2)*	
05	**Software quality Assurance** ***Learning Objective/s:*** *To summarize basics of software quality assurance and management. Also expected to use and analyse various quality factors and criteria which helps to prevent defects from reaching end-users*	**04**
	Task 10: Identify Quality Attributes and Relationships for the various properties used for the selected problem statement. **Task 11:** Implement quality attributes using McCall's quality factors and criteria.	
	Self-Learning Topics: *Web based and Mobile based software quality assurance factors and criteria.* ***Learning Outcomes:*** *A learner will be able to* *LO 5.1: Identify quality attributes for software quality assurance. (P.I.-1.3.1,7.1.1)* *LO 5.2: Apply Software Quality Management and Assurance for to identify possible solution. (P.I.-1.4.1)* *LO 5.3: Apply engineering principles to formulate modules of a system through quality attributes using McCall's quality factors and Criteria. (P.I.-2.3.1,7.2.2,8.2.1,9.2.1)* *LO 5.4: Analyze quality attributes and criteria used for software development using ISO9000:2000, SIX sigma. (P.I.-2.3.2,8.3.1,9.2.2)*	
	Total	**30**

Performance Indicators :

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem
2.1.1	Evaluate problem statements and identifies objectives
2.1.2	Identifies processes/modules/algorithms of a computer-based system and parameters to solve a problem.
2.2.4	Compare and contrast alternative solution/methods to select the best methods
2.2.5	Compare and contrast alternative solution processes to select the best process.
2.3.1	Able to apply computer engineering principles to formulate modules of a system with required applicability and performance
2.3.2	Identify design constraints for required performance criteria
2.4.3	Identify the limitations of the solution and sources/causes

3.1.2	Able to identify and document system requirements from stake holders
3.1.6	Ability to develop software requirement specifications (SRS).
3.2.1	Ability to explore design alternatives.
3.2.2	Ability to produce a variety of potential design solutions suited to meet functional requirements.
3.4.1	Ability to refine architecture design into a detailed design within the existing constraints.
3.4.2	Ability to implement and integrate the modules.
3.4.3	Ability to verify the functionalities and validate the design.
4.1.2	Ability to choose appropriate procedure/algorithm, data set and test cases.
4.3.1	Use appropriate procedures tools and techniques to collect and analyze data
4.3.2	Critically analyse data for trends and correlations, stating possible errors and limitations
5.1.2	Create/adapt/modify/extend tools and techniques to solve engineering problems.
5.3.2	Verify the credibility of results from tool use with reference to the accuracy and limitations, and the assumptions inherent in their use.
6.1.1	Identify risks/impacts in the life-cycle of an engineering product or activity.
6.2.2	Apply principles of preventive engineering and sustainable development to an engineering activity or product relevant to the discipline
8.2.1	Demonstrate effective communication, problem solving, conflict resolution and leadership skills
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts.
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents
9.2.1	Listen to and comprehend information, instructions, and viewpoints of others
9.2.2	Deliver effective oral presentations to technical and non-technical audiences
9.3.2	Use a variety of media effectively to convey a message in a document or a presentation
10.2.1	Analyze and select the most appropriate proposal based on economic and financial considerations.
10.3.1	Identify the tasks required to complete an engineering activity, and the resources required to complete the tasks.

Course Outcomes:

Learner will be able to

1. Use concepts of software engineering process models to optimize the flow of project development process. *(LO 1.1,LO 1.2, LO 1.3,LO 1.4, LO 1.5,LO 1.6)*
2. Identify requirements, analyse, plan, schedule and track the progress of the projects. *(LO 2.1,LO 2.2, LO 2.3,LO 2.4, LO 2.5,LO 2.6)*
3. Use various software testing methods and strategies to identify bugs. *(LO 3.1,LO 3.2, LO 3.3,LO 3.4, LO 3.5,LO 3.6,LO 3.7, LO 3.8)*
4. Illustrate use of automation tools. *(LO 4.1,LO 4.2, LO 4.3,LO 4.4, LO 4.5,LO 4.6,LO 4.7, LO 4.8)*
5. Apply the software quality assurance factors to Identify Quality Attributes and their Relationships. *(LO 5.1,LO 5.2, LO 5.3,LO 5.4)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITLBC405.1	3	3	3	-	-	-	-	-	-	3	-
ITLBC405.2	3	3	3	-	3	-	-	-	-	3	-
ITLBC405.3	3	3	3	3	-	-	-	-	-	-	-
ITLBC405.4	3	3	-	3	3	-	-	-	-	-	-
ITLBC405.5	3	3	-	-	-	3	-	3	3	-	-
Average	**3**	**3**	**3**	**3**	**3**	**3**	**-**	**3**	**3**	**3**	**-**

Text Books:

1. Software Engineering: A practitioner's approach, Roger S. Pressman, 7 th edition ,2010, McGraw Hill
2. Fundamentals of Software Engineering, Rajib Mall, 7 th edition ,2014, Prentice Hall India
3. Software Testing Principles and Practices, Naresh Chauhan, 2 nd edition, 2016, Oxford Higher Education
4. Software Testing and quality assurance theory and practice, Kshirasagar Naik, Priyadarshi Tripathy, 1st edition 2008,Wiley Publication.

Reference Books:

1. An integrated approach to Software Engineering, Pankaj Jalote, , 3rd edition 2005, Springer/Narosa.
2. Effective Methods for Software Testing, Willam E. Perry, third edition, 2006,Wiley Publication.
3. Software Testing Concepts and Tools, Nageswara Rao Pusuluri, ISE Edition, 2006, Dreamtech press

Other Resources:

1. NPTEL Course: Introduction to software Engineering By Prof. N.L.Sarda, Department of Computer Science and Engineering, IIT Bombay
   Web link- https://archive.nptel.ac.in/courses/106/101/106101061/
2. Course on Software Testing By Prof. Rajib mall, Department of Computer Science and Engineering, IIT Kharagpur
   Web link- https://nptel.ac.in/courses/106105150/.

A. CONTINUOUS ASSESSMENT (25 MARKS)

Suggested breakup of distribution

a) Task Execution :10 Marks

Students will be given 11 tasks based on list as per mentioned in the syllabus. Each task carries 10 Marks. Average will be taken of all tasks. Students are expected to identify and apply software engineering principles also perform manual and automated testing to find the logical and structural errors and also apply quality assurance factors and criteria. Students will be evaluated based on parameters such as identification of process models, testing techniques and quality check.

b) Practical Test: 05 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

c) Oral based on entire syllabus: 05 Marks

d) Regularity & Active participation: 05 Marks

B. END SEMESTER EXAMINATION (Practical and Oral Exam) (25 Marks)

Suggested breakup of distribution

a) Task Execution :10 Marks

Students will be given task (different for each student) and they will be evaluated based on the parameters mentioned in continuous assessment.

b) Documentation on Software Development Life Cycle: 05 Marks

c) Oral based on entire syllabus:10 Marks

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
SBL	ITSBL402	FULL STACK DEVELOPMENT LABORATORY	02

Examination Scheme		
Continuous Assessment	**End Semester Exam (ESE)**	**Total**
50	50	100

Pre-requisite :

1. ESL205: Programming Laboratory-II (Java)
2. ITSBL301: Python Laboratory

Program Outcomes addressed :

1. PO1: Engineering knowledge
2. PO2: Problem analysis
3. PO3: Design/Development of solution
4. PO5: Engineering tool usage
5. PO8: Individual and Collaborative team work
6. PO9: Communication
7. PO11: Life-long learning

Course Objectives:

1. To equip students with the skills to design and develop web applications using HTML and multimedia elements.
2. To equip students with the skills to develop and style web applications using HTML and CSS, focusing on creating responsive and interactive user interfaces.
3. To equip students with the skills to analyze and interpret programming results using JavaScript techniques.
4. To equip students with the skills to evaluate and implement modern front-end technologies, focusing on React and AngularJS.
5. To equip students with the skills to install, configure, and develop applications using Node.js, Express, and MongoDB.

Module	Detailed Contents	Hrs
00.	**Course Introduction** In the Full Stack Web Development Lab, students explore modern web development using React and Angular. They gain practical experience building dynamic web applications from frontend to backend.	**01**
01.	**HTML5** ***Learning Objective/s:*** *To demonstrate and apply HTML tags to develop webpages with well-structured HTML. Also expected to create web pages using text formatting, graphics, audio, and video elements.*	**03**

	Contents: Elements, Attributes, Head, Body, Hyperlink, Formatting, Images, Tables, List, Frames, Forms, Multimedia. **Task 1:** Implement using HTML5: Elements, Attributes, Head, Body, Hyperlink, Formatting, Images, Tables, List, Frames, Forms, Multimedia	
	Self-Learning Topics: *Canvas and SVG Graphic, Parallax Scrolling* ***Learning Outcomes :*** *A learner will be able to* *LO 1.1: Apply HTML elements and attributes to structure web pages, utilize head and body tags to organize content, and integrate hyperlinks to connect web resources, ensuring a comprehensive understanding of web page formatting and structure. (P.I.- 1.3.1, 3.4.2, 5.1.1, 8.1.1)* *LO 1.2: Implement multimedia elements such as images, frames, and forms to enhance user interaction, demonstrate proficiency in creating and managing tables and lists for data presentation, and collaborate effectively in teams to develop and debug web applications, ensuring seamless integration of multimedia components. (P.I.- 1.4.1, 3.4.3, 5.2.2, 8.3.1)*	
02.	**CSS3** ***Learning Objective/s:*** *To apply CSS in styling tables and lists, also use advanced selectors to precisely target and style specific elements within a document.*	**04**
	Contents: Syntax, Inclusion, Color, Background, Fonts, Tables, lists, CSS3 selectors, Pseudo classes, Pseudo elements **Task 2:** Implement using CSS3: Syntax, Inclusion, Color, Background, Fonts, Tables, lists, CSS3 selectors, Pseudo classes, Pseudo elements	
	Self-Learning Topics: *CSS Best Practices and Optimization* ***Learning Outcomes :*** *A learner will be able to* *LO 2.1: Apply HTML syntax and inclusion techniques to develop web pages, utilize color and background properties to enhance visual appeal, recognize the strengths and limitations of various font styles, and read, understand, and interpret technical documentation related to HTML and CSS. (P.I.- 1.3.1, 2.1.2, 5.2.1, 9.1.1)* *LO 2.2: Apply CSS3 selectors, pseudo-classes, and pseudo-elements to style web pages, identify and implement tables and lists for data organization, demonstrate proficiency in using CSS for complex styling tasks, and demonstrate effective communication by producing well-structured CSS design documentation. (P.I.- 1.4.1, 2.3.2, 5.2.2, 9.1.2)*	
03.	**Bootstrap** ***Learning Objective/s:*** *To create responsive, mobile-first websites and to ensure all interface elements of a website work optimally on all screen sizes.*	**04**
	Contents: Grid system, Forms, Button, Navbar, Breadcrumb, Jumbotron	

	Task 3: Implement using Bootstrap: Grid system, Forms, Button, Navbar, Breadcrumb, Jumbotron	
	Self-Learning Topics: *Customizing Bootstrap Themes*	
	Learning Outcomes: *A learner will be able to* *LO 3.1: Implement and integrate grid systems and forms within a web application, identify and apply HTML and CSS tools to create responsive and interactive user interfaces, and evaluate the limitations of these tools to validate their effectiveness in real-world applications. (P.I.- 3.4.2, 5.1.1, 5.3.1)* *LO 3.2: Design and develop interactive UI components by integrating buttons, navbars, breadcrumbs, and jumbotrons, and evaluate the limitations of these components, validating their effectiveness in real-world applications. (P.I.- 3.4.3, 5.3.1)*	
04.	**JavaScript** ***Learning Objective/s:*** *To recall and apply JavaScript and its concepts and develop understanding behind JavaScript applications.*	**16**
	Contents: Variables, Operators, Conditions, Loops, Functions, Events, Classes and Objects, Error handling, Validations, Arrays, String, Date **Task 4:** Implement using JavaScript: Variables, Operators, Conditions, Loops, Functions, Events, Classes and Objects. **Task 5:** Implement using JavaScript: Error handling, Validations, Arrays, String, Date.	
	Self-Learning Topics: *Testing and Test-Driven Development, Web APIs and Browser Features* ***Learning Outcomes:*** *A learner will be able to* *LO 4.1: Analyze and interpret programming results using contemporary tools, identify the strengths and limitations of variables, operators, and conditions, and demonstrate proficiency in using discipline-specific tools for handling loops, functions, and events in programming. (P.I.- 2.4.2, 5.2.1, 5.2.2)* *LO 4.2: Identify the limitations of various programming constructs such as classes, objects, and error handling, and demonstrate proficiency in using discipline-specific tools for validations, arrays, strings, and date manipulations. (P.I.- 2.4.3, 5.2.2)*	
05.	**React and Angular** ***Learning Objective/s:*** *To recognize the steps involved in setting up and configuring a React and Angular development environment to ensure a smooth setup. Design and develop an understanding behind React apps and Angular apps.*	**20**
	Contents: Installation and Configuration. JSX, Components, Props, State, Forms, Events, Routers, Refs, Keys. Overview of AngularJS, Need of AngularJS in real web sites, AngularJS modules, AngularJS built-in directives, AngularJS custom directives, AngularJS expressions, Angular JS Data Binding, AngularJS filters, AngularJS controllers, AngularJS scope, AngularJS dependency	

	injection, Angular JS Services, Form Validation, Routing using ng-Route, ng-Repeat, ng-style, ng-view, Built-in Helper Functions, Using Angular JS with Typescript **Task 6:** Implement using React: Installation and Configuration. JSX. Implement using React: Components, Props, State, Forms, Events. Implement using React: Routers, Refs, Keys. **Task 7:** Create a simple HTML "Hello World" Project using AngularJS Framework and apply ng-controller, ng-model and expressions. **Task 8:** Events and Validations in AngularJS. (Create functions and add events, adding HTML validators, using $valid property of Angular, etc.) Create an application for like Students Record using AngularJS	
	Self-Learning Topics: *Context API and Redux, State Management with NgRx (Optional), Testing and Debugging* ***Learning Outcomes:*** *A learner will be able to* *LO 5.1: Evaluate and implement React components, props, and state management in web applications, recognize the importance of lifelong learning in adapting to evolving front-end technologies, and analyze sourced technical information for the feasibility and sustainability of React integration. (P.I.- 3.4.2, 11.1.1, 11.3.2)* *LO 5.2: Verify functionalities and validate the integration of AngularJS services, form validation, and routing, and recognize the importance of lifelong learning in adapting to evolving front-end technologies. (P.I.- 3.4.3, 11.1.1)*	
06.	**Node.js** ***Learning Objective/s:*** *To apply callback functions and event loops, also able to install and configure Express with ease, and build a working Express application on their own to showcase server-side JavaScript coding proficiency. Also expected to build REST API using MongoDB.*	**12**
	Contents: Installation and Configuration, Callbacks, Event loops, creating express app, create Mongodb application **Task 9:** Implement using Node.js: Installation and Configuration, Callbacks, Implement using Node.js: Event loops, Creating express app. Implement Mongodb application. **Task 10:** Implementation of Mini Project on selected case study.	
	Self-Learning Topics: *Template Engines and Views, Indexes and Query Optimization* ***Learning Outcomes:*** *A learner will be able to* *LO 6.1: Identify modern engineering tools for the installation and configuration of Node.js applications, recognize and adapt to diverse working and learning preferences, read, understand, and interpret documentation for callbacks and event loops, and demonstrate proficiency in using discipline-specific tools for creating Express and MongoDB applications. (P.I.- 5.1.1, 5.2.2, 8.1.1, 9.1.1)* *LO 6.2: Present application development results as a team with well-integrated contributions from all members, produce clear, well-constructed, and well-supported documentation for Express and MongoDB applications, and demonstrate proficiency in*	

	using discipline-specific tools for creating and managing these applications. (P.I.- 5.2.2, 8.3.1, 9.1.2, 5.2.2)	
	Total	**60**

Performance Indicators:

P.I. No.	P.I. Statement
1.3.1	Apply engineering fundamentals
1.4.1	Apply theory and principles of computer science engineering to solve an engineering problem
2.1.1	Evaluate problem statements and identifies objectives
2.1.2	Identifies processes/modules/algorithms of a computer based system and parameters to solve a problem
2.2.4	Compare and contrast alternative solution/methods to select the best methods
2.4.2	Analyze and interpret the results using contemporary tools.
2.4.3	Identify the limitations of the solution and sources/causes.
3.4.2	Ability to implement and integrate the modules.
3.4.3	Ability to verify the functionalities and validate the design.
5.1.1	Identify modern engineering tools, techniques and resources for engineering activities
5.2.1	Identify the strengths and limitations of tools for (i) acquiring information, (ii) modeling and simulating, (iii) monitoring system performance, and (iv) creating engineering designs.
5.2.2	Demonstrate proficiency in using discipline specific tools
5.3.1	Discuss limitations and validate tools, techniques and resources
8.1.1	Recognize a variety of working and learning preferences; appreciate the value of diversity on a team
8.3.1	Present results as a team, with smooth integration of contributions from all individual efforts
9.1.1	Read, understand and interpret technical and nontechnical information
9.1.2	Produce clear, well-constructed, and well-supported written engineering documents
11.1.1	Describe the rationale for requirement for continuing professional development
11.3.2	Analyze sourced technical and popular information for feasibility, viability, sustainability, etc.

Course Outcomes:

Learner will be able to

1. To structure web pages, integrate multimedia components, and collaborate effectively in web application development. *(LO 1.1, LO 1.2)*
2. Apply HTML and CSS techniques to build and style web pages, integrate various UI components, and effectively communicate their design through well-structured documentation. *(LO 1.1, LO 1.2, LO 2.1, LO 2.2, LO 3.1, LO 3.2)*
3. Identify the strengths and limitations of JavaScript programming constructs and demonstrate proficiency in using discipline-specific tools for handling loops, functions, events, validations, arrays, strings, and date manipulations. *(LO 4.1, LO 4.2)*

4. Evaluate and implement React components and state management, and verify and validate AngularJS services, form validation, and routing, while recognizing the importance of lifelong learning in adapting to evolving technologies. *(LO 5.1, LO 5.2)*
5. Identify and use modern engineering tools for Node.js applications, collaborate effectively in teams, and produce well-constructed documentation for Express and MongoDB applications. *(LO 5.1, LO 5.2, LO6.1, LO6.2)*

CO-PO Mapping Table with Correlation Level

CO ID	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
ITSBL402.1	3	-	3	-	3	-	-	3	-	-	-
ITSBL402.2	3	3	3	-	3	-	-	3	3	-	-
ITSBL402.3	-	3		-	3	-	-	-	-	-	-
ITSBL402.4	-	-	3	-		-	-	-	-	-	3
ITSBL402.5	-	-	3	-	3	-	-	3	3	-	3
Average	3	3	3	-	3	-	-	3	3	-	3

Text Books:

1. HTML & CSS: Design and Build Web Sites, Jon Duckett,, First Edition, 2011, Wiley.
2. HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery) 2Ed., DT Editorial Services, Second Edition, 2016, Dreamtech Press.
3. Learning React Functional Web Development with React and Redux, Alex Banks and Eve Porcello, First Edition, 2017, Shroff/O'Reilly O'Reilly
4. Learning Node.js Development, Andrew Mead, Kindle Edition, 2018, Packet Publishing

Reference Books:

1. JavaScript & jQuery: Interactive Front-End Development, Jon Duckett, First edition, 2014, Wiley .
2. Full Stack Web Development For Beginners, Riaz Ahmed, 2021
3. Beginning Node.js, Express & MongoDB Development, Greg Lim, 2020
4. Full-Stack React Projects: Modern web development using React 16, Node, Express, and MongoDB, Shama Hoque, First edition, 2018, Packt Publication

Other Resources:

1. JavaScript Tutorial
   Web link: https://www.w3schools.com/js/
2. React: The library for web and native user interfaces
   Web link: https://react.dev/
3. Deliver web apps with confidence
   Web link: https://angular.io/
4. Run JavaScript Everywhere
   Web link: https://nodejs.org/en
5. Express4.18.3 Fast, unopinionated, minimalist web framework for Node.js
   Web link: https://expressjs.com/

6. MongoDB

   Web link: https://www.mongodb.com/

A. CONTINUOUS ASSESSMENT (50 MARKS)

1. Task Execution (30 Marks)

Students will be given minimum 10 tasks.

Students are expected to

i. Identify and apply the appropriate HTML tags to develop a webpage.
ii. Identify and apply the appropriate CSS tags to format data on webpage.
iii. Apply JavaScript to add functionality to web pages.
iv. Use React and Angular to develop the front-end user interface, incorporating components, state management, and routing for seamless navigation and interaction with the back-end API.
v. Construct web based Node.js applications using Express.
vi. Identify the components necessary for building a full stack application, including front-end frameworks like React and Angular, and back-end technologies like MongoDB, Express, and Node.js.
vii. Design a relational database schema using MongoDB for storing application data efficiently.
viii. Implement RESTful API endpoints using Express.js to handle CRUD operations for interacting with the MongoDB database.

Students will be evaluated based on following:

i. Logic building for the given task (10 marks)
ii. Rectifying logical errors and syntax errors (06 marks)
iii. Well-structured and organized program (06 marks)
iv. Effective use and integration of the various technologies involved in full stack development, including React, Angular, MongoDB, Express, and Node.js. (08 marks)

Refer the sample task given below.

Example:

Implement JSX code in React

Students are expected to identify.

i. JSX: Students are expected to identify JSX syntax throughout the components, such as , <h2>, <p>, <ul>, <li>, and <a>, which represent HTML elements. JSX allows mixing HTML-like syntax within JavaScript code in React.
ii. Components: Students should recognize that each component is defined as a function that returns JSX elements. This is a fundamental concept in React - components encapsulate the UI logic and structure.
iii. Props and State: Although not explicitly demonstrated in this example, students can understand that props and state can be passed to components to customize their behavior and appearance.
iv. Hooks: In the BlogPost component, students can identify the usage of the useParams hook from React Router to access URL parameters, demonstrating the use of hooks in React functional components.

v. Router: Students can see the usage of React Router's <Route> component to define routes and their corresponding components, allowing for declarative routing in the application.

By analyzing this code, students can gain a deeper understanding of how JSX syntax is used to define the UI of React components and how React Router facilitates navigation between different views in a React application.

Students will be evaluated based on following:

i. Logic building for the given task (10 marks)
ii. Rectifying logical errors and syntax errors (06 marks)
iii. Well-structured and organized program (06 marks)
iv. Effective use and integration of the various technologies involved in full stack development, including React, Angular, MongoDB, Express, and Node.js. (08 marks)

2. Regularity and active Participation (05 Marks)

3. Mini Project Evaluation (15 Marks)

A group of 3-4 students should be assigned a real life problem statement. Evaluation will focus on the project's ability to meet functional requirements, such as CRUD operations, user authentication, and data validation, ensuring that the application behaves as intended.

Attention will be given to the usability and intuitiveness of the application's interface, including navigation, responsiveness, and feedback mechanisms, to ensure a positive user experience.

Assessment will consider the quality of the codebase, including adherence to coding standards, modularity, reusability, and readability, promoting maintainability and scalability of the project.

Projects will be evaluated for performance optimization techniques, such as minimizing load times, reducing server requests, and efficient database queries, to ensure responsiveness and efficiency.

Evaluation will include the completeness and clarity of project documentation, including setup instructions, user guides, API documentation, and code comments, facilitating understanding and collaboration among developers.

B. END SEMESTER EXAMINATION (Practical & Oral Exam) (50 Marks)

a) Task Execution: 30 Marks
Students will be given task (different task for every student) to execute and will be evaluated as per the parameters mentioned in continuous evaluation

b) Mini Project Presentation: 20 Marks
Students will give mini project presentation and will be evaluated as per the parameters mentioned in continuous evaluation

Two examiners, one Internal and one External will do the evaluation.

Course Type	Course Code	Course Name	Credits
MNP	ITMNP402	MINI PROJECT- 1B	01

Program Outcomes addressed:

1. PO1 : Engineering knowledge
2. PO2 : Problem Analysis
3. PO3 : Design/Development of Solutions
4. PO4 : Conduct investigations of complex problems
5. PO5 : Engineering Tool Usage
6. PO6 : The Engineer & The world
7. PO7 : Ethics
8. PO8 : Individual & Collaborative team work
9. PO9 : Communication
10. PO10: Project Management & Finance
11. PO11: Life-long learning

Course Objectives:

1. To familiarize students about available infrastructure at Department/Institute level, online resources, plagiarism, expectations from MP 1A and 1B, etc.
2. To guide students in identifying societal or research needs and formulating them into problem statements.
3. To facilitate problem-solving in group settings.
4. To apply basic engineering principles to address identified problems.
5. To foster self-learning and research skills.

Course Outcomes:

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

1. Identify problems based on societal or research needs and methodology for solving them.
2. Apply knowledge and skills to solve societal problems collaboratively.
3. Develop interpersonal skills necessary for teamwork.
4. Analyze, verify, and validate results effectively through various methodologies, including, test cases/benchmark data/theoretical/inferences/experiments/simulations, etc.
5. Evaluate the societal and environmental impacts of proposed solutions.
6. Adhere to standard engineering practices.
7. Excel in written and oral communication by technical report writing, oral presentation, and publishing results in
   - Research/white paper/article/blog writing/publication, etc.
   - Business plan for entrepreneurship product creation
   - Patent filing/copyright.
8. Gain technical competencies by participating in competitions, hackathons, etc.
9. Demonstrate lifelong learning capabilities through self-directed group projects.
10. Apply project management principles effectively.

Guidelines for the Mini Project

- At the beginning of semester-III, project guides are required to conduct around 4 hours' orientation sessions including following topics:
  - Familiarizing students about infrastructure available at Department/Institute level and how to use it.

- How to identify societal problems and formulate project problem statement.
- How to carry out literature survey.
- What is plagiarism and what care needs to be taken while writing a report.
- What is project report template and how it should be used.
- What are expectations from mini-projects 1A and 1B.

• Mini project may be carried out in one or more form of following:

Product preparations, prototype development model, fabrication of set-ups, laboratory experiment development, process modification/development, simulation, software development, integration of software (frontend-backend) and hardware, statistical data analysis, creating awareness in society/environment etc.

• Students must form groups of 3 to 4 members either from the same or from different departments.
• Groups should conduct surveys to identify needs and develop problem statements in consultation with faculty.
• An implementation plan in Gantt/PERT/CPM chart format covering weekly activities must be submitted.
• Each group must maintain a logbook to record weekly progress, to be verified by the faculty supervisor.
• Faculty input should emphasize guiding by faculty and self-learning by group members.
• Groups should propose multiple solutions, select the best one in consultation with the supervisor, and develop a working model.
• The solution to be validated with proper justification and report to be compiled in standard format of the Institute. Research papers, competition certificates may be submitted as part of annexure to the report.
• With the focus on self-learning, innovation, addressing societal/research/innovation problems and entrepreneurship quality development within the students through the Mini Projects, it is preferable that a single project of appropriate level and quality be carried out in two semesters by all the groups of the students.
• However, based on the individual students or group capability, with the mentor's recommendations, if the proposed Mini Project adhering to the qualitative aspects mentioned above, gets completed in odd semester, then that group can be allowed to work on the extension of the Mini Project with suitable improvements/modifications or a completely new project idea in even semester. This policy can be adopted on a case by case basis.

In-Semester Continuous Assessment and End-Semester Examination Guidelines

- The Head of the Departments will assign a guide to each of the mini-projects and shall form a progress monitoring committee. The guide will carry out weekly monitoring of the project's progress. The committee shall carry out in-semester project evaluation based on presentations with a minimum of two evaluations per semester.
- Assessment will be based on individual contributions, understanding, and responses to questions asked.
- **Continuous Assessment marks distribution in semester III (50 marks):**

- 10 marks for the Topic Approval Presentation in front of the progress monitoring committee
- 15 marks for the Mid-Semester Progress Presentation in front of the progress monitoring committee
- 20 marks for the Final Report & Presentation
- 05 marks for Regularity and active participation

- **Continuous Assessment marks distribution in semester IV (50 marks):**
  - 15 marks for the In-Semester Two Presentations
  - 10 marks for the Participation in Project Competitions, TPP, etc.
  - 20 marks for the Final Report & Presentation
  - 05 marks for Regularity and active participation

The review/progress monitoring committee will assess projects based on the following criteria.

Semester III:

- Theoretical solution completion, including component/system selection/design of software solution and cost analysis.
- Two reviews will occur:
  - The first review will focus on finalizing the problem statement (topic approval).
  - The second review will focus on finalizing the proposed solution.

Semester IV:

- Expected tasks include procuring components/systems, constructing a working prototype, and validating results based on prior semester work.
- Reviews will be conducted as follows:
  - The first review will assess the readiness to build a working prototype.
  - The second review will involve a poster presentation and demonstration of the working model in the last month of the semester.

In addition to the above-mentioned points, the following performance criteria shall be included during the in-semester continuous assessment:

1. Quality of survey and need identification.
2. Clarity and innovativeness in problem definition and solutions.
3. Requirement gathering feasibility study, cost-effectiveness, and societal impact of proposed solutions.
4. Completeness and full functioning of the working model.
5. Effective use of skill sets and engineering norms.
6. Verification & validation of the solutions/test cases.
7. Individual contributions to the group.

8. Clarity in written and oral communication.
9. Participation in technical paper presentations/project competitions/hackathon competitions, etc.

End-Semester Examination in Semester IV (50 marks):

1. Presentation and demonstration to internal and external examiners: 20 marks.
2. Emphasis on problem clarity, innovativeness, societal impact, functioning of the model, skill utilization, and communication clarity: 30 marks.

Course Type	Course Code	Course Name	Credits
VEC	VEC402	ENVIRONMENT & SUSTAINABILITY	02

Program Outcomes addressed:

1. PO2 : Problem Analysis
2. PO6 : The Engineer & The world
3. PO7 : Ethics
4. PO11: Life-long learning

Course Objectives :

1. To provide students with foundational knowledge and understanding of environmental science principles and concepts.
2. To explore the principles of sustainability and their applications in various domains of engineering and technology.
3. To familiarize students with the legal and ethical considerations associated with environmental management and sustainability practices.
4. To equip students with practical skills and strategies for promoting renewable energy, energy efficiency, waste management, and environmental impact assessment.

Module	Details
01.	**Foundations of Environmental Sciences** Introduction to Environmental Science, Earth's Systems: Atmosphere, Hydrosphere, Lithosphere, Biosphere, Ecological Principles: Energy flow, Nutrient cycling, Biodiversity, Environmental Degradation: Pollution, Deforestation, Habitat loss, Environmental Monitoring and Data Analysis.
02.	**Sustainability Basics** Concepts of Sustainability and Sustainable Development, Sustainable Resource Management: Water, Air, Land, Sustainable Agriculture and Food Systems, Sustainable Transportation and Urban Planning, Sustainable Business Practices and Corporate Social Responsibility
03.	**Legal & Ethical Considerations** Environmental Laws and Regulations: National and International Perspectives, Environmental Policies and Governance Frameworks, Ethical Issues in Environmental Decision Making, Environmental Justice and Equity, Corporate Ethics and Environmental Responsibility
04.	**Renewable energy & Energy efficiency** Introduction to Renewable Energy Sources: Solar, Wind, Hydro, Biomass, Geothermal, Energy Conversion Technologies and Systems Energy Efficiency Measures and Strategies, Policy Support for Renewable Energy Deployment, Economic and Environmental Impacts of Renewable Energy

05.	**Waste management & recycling** Solid Waste Management: Collection, Treatment, Disposal, Recycling Processes and Technologies, E-waste Management and Hazardous Waste Handling, Circular Economy Principles, Waste Reduction Strategies: Source Reduction, Reuse, Repair
06.	**Environmental Impact Assessment** Introduction to Environmental Impact Assessment (EIA), EIA Process: Screening, Scoping, Impact Assessment, Mitigation, Monitoring, Methods and Tools for Impact Assessment: GIS, LCA, Risk Assessment, Case Studies of EIA in Various Sectors: Infrastructure, Energy, Mining, Construction, Role of Stakeholders in EIA Process
Total no. of hours: **30**	

Course Outcomes :

1. Gain a comprehensive understanding of key environmental science principles and their relevance to engineering disciplines.
2. Apply principles of sustainability to analyze and address environmental challenges in engineering projects and processes.
3. Demonstrate awareness of legal and ethical considerations in environmental decision-making and management practices.
4. Develop proficiency in implementing renewable energy technologies and energy-efficient practices in engineering designs and operations.
5. Acquire knowledge and skills in waste management, recycling, and circular economy principles for sustainable resource utilization.
6. Apply environmental impact assessment methods to evaluate and mitigate the environmental impacts of engineering projects and activities.

Text Books :

1. Environmental Science: Toward a Sustainable Future by Richard T. Wright and Dorothy F. Boorse (Publisher: Pearson Education)
2. Introduction to Environmental Engineering and Science by Gilbert M. Masters and Wendell P. Ela (Publisher: Pearson Education)
3. Renewable and Efficient Electric Power Systems by Gilbert M. Masters (Publisher: Wiley)

Reference Books :

1. Environmental Law Handbook by Thomas F. P. Sullivan, David R. Buente Jr., and Sally Fairfax, Bernan Press
2. Sustainability Science by Bert J. M. de Vries, Springer
3. Environmental Impact Assessment: Theory and Practice by Peter Wathern, Routledge

Other Resources:

1. NPTEL Course: Introduction to Environmetal Engineering & Science- Fundamental & Sustainability Concepts, Prof.Brajesh Kumar Dubey, Department of Multidisciplinary IIT Kharagpur :-Web link https://archive.nptel.ac.in/courses/127/105/127105018/

2. NPTEL Course: Environment And Development, By Prof. Ngamjahao Kipgen, IIT Guwahati, Web link- https://onlinecourses.nptel.ac.in/noc23_hs133/preview