Shri Vile Parle Kelavani Mandal's

SVKM NMIMS Global University, Dhule

School of Technology, Management & Engineering

COMPUTER SCIENCE & ENGINEERING (Data Science)

SEMESTER - III SYLLABUS

Shri Vile Parle Kelavani Mandal's

SVKM NMIMS Global University, Dhule

School of Technology, Management & Engineering

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Computational Mathematics				Code: T25CCM201	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
4	3	-	1	40	60	100

Prior knowledge of Engineering Mathematics-I, Engineering Mathematics-II

Course Objectives:

This course aims to enable students to:

1. **Apply** matrix theory concepts to solve engineering and computational problems.
2. **Apply** linear algebra methods for solving engineering problems.
3. **Apply** Fourier transform techniques in signal-processing and computational applications.
4. **Apply** Z-transform methods to solve discrete-time system problems.
5. **Apply** linear programming techniques to solve optimization problems.

Course Outcomes:

After successful completion of the course, the students will be able to:

CO1: Apply matrix theory concepts such as Cayley–Hamilton theorem, eigenvalues and eigenvectors, matrix diagonalization, and functions of square matrices to solve engineering and computer science problems.

CO2: Apply linear algebra techniques including inner product spaces, orthogonality, Gram–Schmidt process, LU decomposition, and singular value decomposition to solve computational and engineering problems.

CO3: Apply Fourier integral theorem, Fourier sine and cosine integrals, Fourier and inverse Fourier transforms, their properties, and Parseval's identity to solve signal-processing and computational problems.

CO4: Apply Z-transform techniques, including inverse Z-transform and convolution theorem, to solve discrete-time computational and signal-processing problems.

CO5: Apply linear programming and optimization techniques such as simplex method and Big-M method to solve optimization problems.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Linear Algebra-I** 1.1 Cayley-Hamilton theorem (without proof) & it's applications 1.2 Eigen values and Eigen vectors, 1.3 Similar matrices, diagonalizable of matrix & it's applications. .	8
II	**Linear Algebra-II** 2.1 Inner product spaces, Orthogonality in R^n, orthonormal set 2.2 Gram-Schmidt Process, 2.3 LU decomposition, 2.4 Singular value decomposition & it's applications to image processing	7

III	**Fourier Transform** 3.1 Fourier Integral theorem, Fourier Sine Integral and Fourier cosine Integral 3.2 Fourier transform and inverse Fourier transform, 3.3 Fourier sine and cosine transforms, Properties of Fourier transform (without proofs) 3.4 Application of Parseval's identity to evaluate integrals	8
IV	**Z-Transform** 4.1. Z-Transform with examples 4.2 Standard properties of Z-transform (without proof), Convolution theorem (without proof). 4.3 Inverse Z-Transform 4.4 Applications of Z-transform	7
V	**Optimization Techniques:** 5.1 Types of solution, Standard and Canonical form of LPP, Basic and feasible solutions. 5.2 Simplex method 5.3 Big –M method (method of penalty).	9
Total		**39**

Text Books:

1. Linear Algebra and its Applications by David C. Lay, Pearson Third Edition
2. Higher Engineering Mathematics by B. V. Ramana, Tata McGraw-Hill Publications, New Delhi.
3. Operation Research by Hira & Gupta, S Chand.

Reference Books:

1. Advanced Engineering Mathematics by Erwin Kreyszig, John Wiley & Sons, New York.
2. Higher Engineering Mathematics by H. K. Das and Er. Rajnish Verma, S. Chand & CO. Pvt. Ltd., New Delhi.
3. Higher Engineering Mathematics by B. V. Ramana, Tata McGraw-Hill Publications, New Delhi.
4. A Text Book of Engineering Mathematics by Peter O' Neil, Thomson Asia Pte Ltd., Singapore.
5. Operations Research by S.D. Sharma Kedar Nath, Ram Nath& Co. Meerat.
6. Engineering optimization (Theory and Practice) by SingiresuS.Rao, New Age International publication

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Computational Mathematics Tutorial				Code: T25CCM201T	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
01	-	-	01	25	-	25
Minimum thirteen tutorials based on above syllabus will be conducted. Mini project relevant to the subject may be included, which would help the learner to apply the concept learnt.						

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Data Structure				Code: T25CDS202	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
03	03	-	-	40	60	100

Prior knowledge of

Course Objectives:

This course aims to enable students:

1. To understand different types of data structures and the concept of Abstract Data Types (ADT).
2. To learn stacks, queues, and linked lists and apply them in solving practical problems.
3. To understand trees and their different types for organizing and processing data efficiently
4. To learn searching techniques and graph concepts, including traversals

Course Outcomes:

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

CO1 Apply appropriate data structures and algorithms to solve problems efficiently
CO2 Apply linear data structures and searching and sorting methods for efficient data processing.
CO3 Apply stack and queue data structures for efficient expression evaluation, recursion, and data processing.
CO4 Apply the concepts of Linked Lists for problem solving
CO5 Apply the concepts of Trees and Graph for problem solving

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction to Data Structures** **Types of Data Structures** - Linear & Nonlinear, Static & Dynamic; Algorithms: Characteristics of Algorithms; Time and Space Complexity. Hashing: Concepts - Hash table, issues in hashing, Hash functions-properties of good hash function, Hash functions: division, Structure extraction, mid-square, folding and universal methods; Collision resolution strategies-open addressing and chaining, Skip List, Introduction of Algorithm Design strategy Divide and conquer, Greedy Approach	7
II	**Unit 02 Linear Data Structures, Searching and sorting** Overview of Array, Array as an Abstract Data Type, Operations on Array, Storage Representation, Multidimensional Arrays, Sparse matrix representation using 2D Searching, Sequential Search/Linear Search, Binary Search, Indexed Sequential Search.	8

	Sorting: Sorting: Concepts- Stability, Efficiency, and Number of Passes, Internal and External Sorting, Bubble sort, Insertion Sort, Selection Sort , Quick Sort, Merge sort	
III	**Stack and Queue** **Stack:** Stack as an ADT; Stack representation using array and linked list; Operations on Stack. Application of stack for expression evaluation and expression conversion, Recursion. **Queue:** Queue as an ADT, Queue representation using array and linked list; Types of Queues: Linear Queue, Circular Queue, Double Ended Queue; Operation on Queue. Priority Queue and its advantages and applications	8
IV	**Linked List** Dynamic Memory management, Basics of Linked List, Comparison of sequential and linked organizations, Types of linked list: Singly linked list, doubly linked list, Circular linked list; Various operations insertion, deletion, searching, updating and printing for linked list.	8
V	**Trees and Graph** **Trees:** General tree and its representation: sequential and linked organization, Binary tree- prop erties, converting tree to binary tree, binary tree traversals - inorder, preorder, post order, Operations on binary tree. Binary Search Tree (BST) and its operation. **Graph:** Basic Concepts, Storage representation, Adjacency matrix, adjacency list, Traversals-depth first and breadth first, Minimum spanning Tree	8
Total		**39**

Text Books:

1. S. Lipschutz, Data Structures, McGraw-Hill Publication, Revised 1st Edition, 2014.
2. Data structures and algorithms made easy, CareerMonk Publications, 5th Edition

Reference Books:

1. Horowitz and Sahani, Fundamentals of Data Structures, Universities Press, 2nd Edition, 2008.
2. Thomas Cormen, Introduction to Algorithms, PHI Publication, 2nd Edition, 2002.

e-sources: NPTEL/SWAYAM Course (if any)

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

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Discrete Mathematics				Code: T25CDM203	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
03	03	-	-	40	60	100

Prior knowledge of

Course Objectives:

This course aims to enable students:

1. To cultivate clear thinking and creative problem-solving.
2. To thoroughly train in the construction and elaboration of mathematical proofs.
3. To Exercise common mathematical arguments and proof strategies.
4. To thoroughly prepare for the mathematical aspects of other Computer Engineering courses.

Course Outcomes:

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

1. To apply the principles of classical and fuzzy set theory, logical reasoning, and proof techniques to model, analyze, and solve mathematical and computational problems.
2. To analyze and construct discrete mathematical models using binary relations, partial orders, lattices, and a variety of function types
3. To model and solve discrete problems using linear recurrence relations, generating functions for sequences and coefficient extraction, and advanced counting techniques.
4. To apply principles and concepts of graph and tree theory in practical situations.
5. To interpret and apply fundamental algebraic structures by using groups and ring–field concepts to model and solve problems in number theory and coding-related contexts.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Set Theory:** Introduction to Set Theory, Venn diagrams, Operations on Sets, Power sets, Laws of set theory, Cartesian Product, Partitions of sets, **Proofs: Proof Techniques, Mathematical Induction** **Introduction to Propositional Logic:** Propositions and Logical operations, Truth tables, Laws of Logic, Equivalence, Implications, Normal Forms, Predicates, Quantifiers**,** First order logic. **Introduction to Fuzzy sets:** Properties of Fuzzy sets, Fuzzy set operations, Fuzzy Cartesian product	8

II	**Relations, Functions, Recurrence Relations:** Definitions, Properties of Binary Relations, Equivalence Relations and partitions, **Posets:** partial ordered relations, Poset, Hasse diagram, Lattice and its types-distributive, complimentary. **Functions:** Definition, Domain, Range, Image, etc. Types of functions: Surjection, Injection, Bijection, Inverse, Identity, Composition of Functions.	7
III	**Recurrence Relation:** linear recurrence relation with constant coefficients, Homogeneous and non-homogeneous recurrence relation, **Generating function**: Function of Sequences, Generating functions, Calculating the Coefficient of the Generating Function. **Counting:** Basic Counting Techniques (sum, product, subtraction, division, exponent), Pigeonhole and Generalized Pigeonhole Principle with many examples, Permutations and Combinations and numerical problems, The Principle of Inclusion and Exclusion**,** Binomial Coefficients Pascal's, Identity and Triangle	10
IV	**Graphs :** Definitions, Paths, circuits, connectivity, Types of Graphs, Eulerian and Hamiltonian Graph, Sub Graphs, Planar Graphs, Chromatic number, Graph coloring. **Functions and Graphs:** Isomorphism of graphs **Introduction to Trees**: Trees, rooted trees, path length in rooted trees, Prefix codes and optimal prefix codes	6
V	**Number Theory and Algebra Groups**: Binary operations, Group, Semigroup, Monoid, Sub-group, and Cyclic group, Homomorphism and Isomorphism of groups, Cosets, Codes and Group codes. **Rings and Fields:** Definition, Sub rings, Integral domain, Field, Integer modulo n, Ring homomorphism.	8
Total		**39**

Text Books:

1. Kenneth H. Rosen, "Discrete Mathematics and Its Applications", 7th Edition, Tata McGraw-Hill, 2017, ISBN: 9780073383095.
2. C. L. LIU, "Elements of Discrete Mathematics", 4th Edition, Tata McGraw-Hill, 2017, ISBN-10: 1259006395 ISBN-13: 978125 9006395.
3. Douglas B West.," Introduction to Graph Theory" 2nd Edition, Eastern Economy Edition published by PHI Learning Pvt. Ltd.
4. Ralph Grimaldi, "Discrete and Combinatorial Mathematics" 5th ed, Pearson Education

5. S.N. Sivanandam, S. N. Deepa, "Principles of Soft Computing", 2nd Edition, 2011 Wiley India Pvt. Ltd

Reference Books:

1. Y N Singh, "Discrete Mathematical Structures", Wiley-India.
2. G. Shanker Rao, "Discrete Mathematical Structures", 2nd Edition2009, New Age International, ISBN-10: 8122426697, ISBN-13: 9788122426694
3. Lipschutz, Lipson, "Discrete Mathematics", 2nd Edition, 1999, Tata McGraw-Hill, ISBN: 007 463710X.
4. V. K. Balakrishnan, "Graph Theory", 1st Edition, 2004, Tata McGraw-Hill, ISBN-10: 007-058718-3, ISBN-13: 9780070587182.

e-sources: NPTEL/SWAYAM Course (if any)

Program:	**B. Tech. (Computer Engineering/DS/AIML)**				**Semester: I**	
Course:	**Discrete Mathematics tutorial**				**Code: T25CDM203T**	
Credit	**Teaching Scheme (Hrs./Week)**			**Evaluation Scheme and Marks**		
	Lecture	**Practical**	**Tutorial**	**ISE**	**ESE**	**Total**
01	**-**	**-**	**01**	**25**	**-**	**25**

Minimum thirteen tutorials based on above syllabus will be conducted. Mini project relevant to the subject may be included, which would help the learner to apply the concept learnt.

1.

1.

1.

Program:	B. Tech. (DS & AIML)				Semester: III	
Course:	Artificial Intelligence-I				Code: T25DAI204/T25AAI204	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
03	03	-	-	40	60	100

Prior knowledge of Mathematical Subjects

Course Objectives:

This course aims at enabling students:

6. To introduce the fundamental concepts, evolution, goals, and characteristics of Artificial Intelligence, including its major branches, real-world applications, and key challenges, enabling students to understand the scope and impact of AI.
7. To introduce the concept of intelligent agents, focusing on agent–environment interaction, types of agents, performance measures, and the principles of rationality and autonomy.
8. To introduce problem formulation and state-space representation and to study uninformed and informed search techniques, including heuristic-based methods, for solving AI problems.
9. To provide hands-on understanding of intelligent systems by designing agent-based models using NetLogo, visualizing search algorithms using Graphviz, and analyzing bias and fairness in AI systems using tools like IBM AI Fairness 360 and Google What-If Tool.
10. To introduce the ethical, social, and future-oriented aspects of Artificial Intelligence, including issues of bias, fairness, privacy, explainability, emerging AI trends, and their societal impact.

Course Outcomes:

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

2. To explain the core concepts of Artificial Intelligence, differentiate AI from human intelligence, identify major AI branches and applications, and discuss the challenges and limitations of AI systems.
3. To explain the functioning of intelligent agents, classify different agent types, and analyze agent behavior based on environment characteristics, performance measures, and rational decision-making.
4. To formulate AI problems as state-space models, apply BFS, DFS, Uniform Cost, Best-First, and A* search algorithms, and analyze their use in real-world AI applications.
5. To develop and simulate intelligent agents, visualize and compare search strategies (BFS, DFS, heuristic and non-heuristic), and evaluate as well as mitigate bias in AI models using standard analytical tools.
6. To explain ethical and privacy challenges in AI, identify bias and transparency issues, describe the importance of Explainable AI, and discuss future AI trends and their impact on society and the workforce.

Detailed Syllabus		
Unit	**Description**	**Duration (H)**
I	**Introduction to Artificial Intelligence:** What is Artificial Intelligence? History and evolution of AI, Goals and characteristics of AI systems, Strong AI vs Weak AI, AI vs Human Intelligence, Branches of AI: ML, NLP, Robotics, Expert Systems, Applications of AI in healthcare, finance, education, industry, Challenges and limitations of AI	8
II	**Intelligent Agents:** Concept of Intelligent Agents, Agent–Environment interaction, Types of agents: Simple reflex, Model-based, Goal-based, Utility-based, Learning agents, Performance measures, environments, Rationality and autonomy	8
III	**Problem Solving and Search Techniques:** Problem formulation and state-space representation, Uninformed search techniques: Breadth First Search (BFS), Depth First Search (DFS), Uniform Cost Search, **Informed search techniques:** Heuristic functions, Best First Search, A* Algorithm, Applications of search in AI	8
IV	**Agent-based modeling using NetLogo :** Designing simple reflex agent simulations ,Modeling goal-based agents,Simulation of multi-agent systems, Observing emergent behavior,**Search tree visualization using Graphviz Tool:** Visualizing BFS and DFS trees,Comparing heuristic vs non-heuristic search,Path cost representation, **Bias and fairness analysis using Tools** as IBM AI Fairness 360 ,Google What-If Tool	8
V	**Ethics, Bias, and Future of AI:** Ethical issues in AI; Bias, fairness, and transparency, Privacy and security concerns, Explainable AI (XAI) – basic idea, Future trends in AI: Human-centred and responsible AI, AI in automation and decision support, Edge AI, generative AI, and autonomous systems (overview) ; AI and societal impact: Impact of AI on employment and workforce, AI in governance, education, and healthcare, Social, economic, and cultural implications of AI	8
Total		**39**

Text Book:

1. Stuart Russell & Peter Norvig, Artificial Intelligence: A Modern Approach, Pearson
2. Elaine Rich & Kevin Knight, Artificial Intelligence, McGraw-Hill

Reference Books:

1. George F. Luger, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson
2. Patrick Henry Winston,Artificial Intelligence, Addison-Wesley
3. Virginia Dignum, Responsible Artificial Intelligence, Springer

e-sources:

1. YouTube: NetLogo Tutorials (search "NetLogo agent-based modeling") ,Hands-on demos for simulation design

2. Graphviz DOT Language Guide

3.BFS/DFS visualization tutorials (step-by-step tree generation)

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Employability Skills Development				Code: T25CED205	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
02	02	-	-	40	60	100

Pre-requisite: Effective Communication Skills

Course Objectives:

This course aims at enabling students:

1. **To** develop students' soft skills and professional communication abilities required for effective interaction in academic, workplace, and societal environments.
2. To familiarize students with the principles, processes, and ethical dimensions of business communication and enable them to communicate effectively through presentations, negotiations, persuasion, and cross-cultural interactions in professional environments.
3. **To** enhance quantitative aptitude, logical reasoning, and verbal ability necessary for solving analytical and placement-oriented problems.
4. **To** strengthen proficiency in written English by improving grammar, comprehension, and professional writing skills.
5. **To** prepare students for employment selection processes by developing interview skills, group discussion competence, active listening, and non-verbal communication.

Course Outcomes:

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

CO1: Demonstrate effective use of soft skills and professional communication techniques to develop resumes, cover letters, presentations, and technical documents for employability.

CO2: Apply business communication principles to plan and deliver effective professional presentations and persuasive interactions in organizational contexts.

CO3: Utilize arithmetic, quantitative, logical reasoning, and verbal aptitude techniques to solve aptitude and analytical reasoning problems efficiently.

CO4: Employ grammatical rules and comprehension strategies to construct, transform, summarize, and convey ideas through clear and professional written English.

CO5: Practice interview, group discussion and non-verbal communication skills to perform effectively in professional selection and workplace interactions.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Soft Skills & Communication basics:** Soft skills Vs hard skills, Skills to master, Interdisciplinary relevance, Global and national perspectives on soft skills, Resume, Curriculum vitae, How to develop an impressive resume, Different formats of resume Chronological, Functional, Hybrid, Job application or cover letter.	5

II	**Fundamentals of Business Communication** Meaning and scope of Business Communication, Process and models of communication, Barriers to communication and strategies to overcome them, Ethics in business communication, Cross-cultural communication basics Professional presentation- planning, preparing and delivering presentations. Negotiation and persuasion basics	6
III	**Arithmetic and Mathematical Reasoning and Analytical Reasoning and Quantitative Ability:** Aspects of intelligence: multiple intelligence theory, Number sequence test, mental arithmetic (square and square root, LCM and HCF, speed calculation, reminder theorem). Matching, Selection, Arrangement, Verifications (Exercises on each of these types). Verbal aptitude (Synonym, Antonym, Analogy).	5
IV	**Grammar and Comprehension:** English sentences and phrases, Analysis of complex sentences, Transformation of sentences, Paragraph writing, Story writing, Reproduction of a story, Letter writing, précis writing, Paraphrasing and e-mail writing.	5
V	**Skills for interviews:** Interviews- types of interviews, preparatory steps for job interviews, interview skill tips, Group discussion- importance of group discussion, types of group discussion, difference between group discussion, panel discussion and debate, personality traits evaluated in group discussions, tips for successful participation in group discussion Non-verbal communication-body movement, physical appearance, verbal sounds, closeness, time.	5
Total		**26**

Text Books:

1. R. Gajendra Singh Chauhan, Sangeeta Sharma, ―Soft Skills- An integrated approach to maximize personality‖, ISBN: 987-81-265-5639-7, First Edition 2016

Reference Books:

1. **Bovee, Courtland L. & Thill, John V.** *Business Communication Today.* Pearson Education, India
2. Wiley Wren and Martin, "English grammar and Composition", S. Chand publications.
3. R. S. Aggarwal, "A modern approach to verbal reasoning", S. Chand publications.
4. Philip Carter, "The Complete Book of Intelligence Test", John Willey & Sons Ltd.

5. Philip Carter, Ken Russell, "Succeed at IQ test", Kogan Page.
6. Eugene Ehrlich, Daniel Murphy, "Schaum;s Outline of English Grammar", McGraw Hills.
7. David F. Beer, David A. McMurrey, ―A Guide to Writing as an Engineer‖, ISBN: 978-1-118-30027-5 4th Edition, 2014, Wiley.

e-sources:

1. NPTEL: Employment Communication A Lab based course, Prof. Seema Singh, IIT Kharagpur.

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Universal Human Values-II				Code: T25CUH206	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
2	2	0	0	40	60	100

Prerequisites: Indian Knowledge System, IPDC

Course Objectives:

This course aims at enabling students:

1. Understand the need, purpose, and process of value education, and explore the concepts of happiness, prosperity, and self-exploration
2. Develop clarity about the human being as a coexistence of the self and the body, and understand harmony within oneself.
3. Appreciate harmony in the family, society, and nature, leading to responsible and value-based human relationships.
4. Recognize the interconnectedness and holistic nature of existence, enabling sustainable and ethical living.
5. Apply holistic understanding to professional ethics, developing competence for value-based decisions and practices in personal and professional life.

Course Outcomes:

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

1. Explain the concepts of value education, self-exploration, happiness, prosperity, and methods to fulfil human aspirations.
2. Differentiate between the needs of the self and the body and demonstrate understanding of harmony within self and with the body
3. Analyze family and societal relationships based on trust and respect, contributing to harmonious living
4. Interpret harmony in nature and existence through the principles of interconnectedness, mutual fulfilment, and coexistence.
5. Exhibit ethical behaviour and apply holistic human values in professional situations, supporting sustainable and humanistic practices

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction to Value Education** Understanding Value Education, Self-exploration as the process for value education, Continuous happiness and prosperity as basic human aspirations, Right understanding, relationship, and physical facility, Happiness and prosperity: current scenario Methods to fulfil basic human aspirations	5

II	**Harmony in the Human Being** Human being as coexistence of the self and the body, needs of the self vs needs of the body, Body as an instrument of the self, Harmony in the self, Harmony of self with the body, Programmed for self-regulation and health	5
III	**Harmony in the Family and Society** Family as the basic unit of human interaction, Values in human-to-human relationship, Trust as the foundational value, Respect as right evaluation, Harmony in society Vision of universal human order	5
IV	**Harmony in Nature (Existence)** Understanding harmony in nature, Interconnectedness, self-regulation, and mutual fulfilment, Four orders of nature, Existence as coexistence at all levels, Holistic perception of harmony in existence	5
V	**Implications of Holistic Understanding – Professional Ethics** Natural acceptance of human values, Definitiveness of ethical human conduct, Humanistic education and universal human order, Competence in professional ethics, Holistic technologies and production systems, Management models and case studies, Transition strategies towards value-based life and profession	6
Total		**26**

Text Books:

1. Human Values and Professional Ethics *R.R. Gaur, R. Sangal, G. Bagaria*
2. A Foundation Course in Human Values and Professional Ethics *R.R. Gaur, R. Sangal, G. Bagaria*

Reference Books:

1. Education for Human Values – From Personal to Global *R. Ramesh*
2. Professional Ethics *Mike Martin & Roland Schinzinger*
3. Ethics in Engineering *Charles B. Fleddermann*
4. Universal Human Values : A Global Perspective on Shared Principles and Ethics" *Dr. Narayan R. Chandak & Adv. Jyotsna N. Chandak*

e-sources: NPTEL/SWAYAM Course (if any)

1. Exploring Human Values: Visions of Happiness & Perfect Society http://nptel.ac.in/courses/109104068
2. https://uhv.org.in/UHV-II_Course_Material

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Data Structure Laboratory				Code: T25CDS202L	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
1	-	2	-	25	25	50

Prior knowledge of

Course Objectives:

This course aims at enabling students:

1. To provide hands-on experience in **implementing core data structures and algorithms** such as hashing, sorting, stacks, queues, linked lists, trees, and graphs.
2. To develop the ability to **apply appropriate data structures and algorithmic techniques** for solving computational problems involving data organization, searching, and traversal.
3. To enhance problem-solving skills by **analyzing and comparing algorithm performance**, focusing on efficiency, correctness, and practical implementation using structured programming.

Course Outcomes:

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

CO1: Implement fundamental **data structures and algorithms** such as hashing, sorting, stacks, queues, linked lists, trees, and graphs using programming techniques.

CO2: Analyze and compare **algorithmic solutions** based on time and space complexity to select efficient approaches for problem solving.

CO3: Apply **stack and queue concepts** to solve practical problems including expression evaluation, expression conversion, and data manipulation.

CO4: Develop programs using **non-linear data structures** like binary trees, binary search trees, and graphs, performing operations and traversals effectively.

CO5: Demonstrate **problem-solving and programming skills** by selecting appropriate data structures for given applications and implementing them efficiently.

Lis of Experiments

Sr. No.	Description
1	Write a program to implement hashing technique using different collision resolution methods. i) Separate Chaining ii) Open addressing
2	Write a menu driven program to implement sorting algorithms: i) Insertion Sort ii) Merge Sort iii) Quick sort iv) Heap Sort.
3	Write a program to implement stack using array.
4	Write a program to evaluate a given postfix expression using stacks.
5	Write a program to convert a given infix expression to postfix expression using stack.

6	Write a program to implement circular queue using array.
7	Write a program to implement double ended queue (dequeue) using arrays.
8	Write a program to implement queue using two stacks.
9	Write a program to implement Single linked list and double linked List with insertion, deletion, searching and traversal operations.
10	Write a program to implement a stack using a linked list such that push and pop operations of stack still take O (1).
11	Write a program to implement binary tree with recursive traversals (inorder, preorder, postorder)
12	Write a program to create binary search tree (BST) to perform insertion, deletion, searching, finding minimum and maximum keys.
13	Write a program to implement graph traversals by applying i) BFS ii) DFS.
Text Book:	
1.	Weiss, Data structures and algorithms analysis in C++, Pearson Education, 4th Edition,2013
Reference Books:	
1.	S. Lipschutz, Data Structures, McGraw-Hill Publication, Revised 1st Edition, 2014.
2.	Y. Langsm, M. Augenstin, A. Tanenbaum, Data Structure using C and C++, Prentice Hall India Learning Private Limited,2nd edition,1998

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Web Technology Lab				Code: T25CWT207	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
01	-	2	-	25	25	50

Prerequisites

- Basic programming (like C/Java), understanding of internet concepts (HTTP, browsers)

Course Objectives:

This course aims at enabling students:

1. To introduce fundamental concepts of web development, website design principles, and the web development life cycle.
2. To enable students to design structured and responsive web pages using HTML, CSS, and Bootstrap.
3. To develop client-side scripting skills using JavaScript and jQuery for dynamic web interaction.
4. To familiarize students with server-side scripting using PHP and database connectivity using MySQL.
5. To provide hands-on experience in building complete web applications integrating frontend and backend technologies.

Course Outcomes:

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

CO1: To Demonstrate web development concepts, HTML structure, and website design principles for creating static web pages.
CO2: To Apply CSS and Bootstrap techniques to design responsive, visually appealing web interfaces.
CO3: To Implement client-side scripting using JavaScript and jQuery to create interactive and dynamic web pages.
CO4: To Develop server-side applications using PHP to process forms, manage sessions, and handle user input securely.
CO5: To Implement PHP with MySQL to perform database operations and develop a basic data-driven web application

Detailed Syllabus

Unit	Description	Duration (H)
I	**Unit I: Introduction to Web Development** **Web Site development Essentials**: Overview of Web Design Concepts, Web Project Management Fundamentals, Web Site Development Process, HTML and the Evolution of Markup languages, HTML basic tags, Web Page Layout and Elements, Create Hyperlinks, Create Tables, Create Web Forms, Image Inserting Techniques, Create Frames, GUI HTML Editors, Site Content and Metadata	5

II	**Unit II: CSS & Responsive Design** **Cascading Style Sheets**: Cascading Style Sheets for Web page design, Creating CSS rules, Format Text with CSS, Use of CSS Selectors, Embed Style Sheets, and Attach External Style Sheets. Using CSS with Tables: Insert and Styling Tables, Import Table Data, Style Tables with CSS, Sort Data in Table,	5
III	**Unit III : Bootstrap:** Introduction to Bootstrap, Bootstrap Basics, Bootstrap Grids, Bootstrap Themes, Bootstrap CSS, Bootstrap JS.	5
IV	**Unit IV: Client Side Scripting** **Introduction to JavaScript**, Variables, Basic in JavaScript — Numbers and operators, Handling text — Strings in JavaScript, Useful string methods, Arrays, Troubleshooting JavaScript; Programming fundamentals: If...Else Statements, Else...If Statements, For Loops, While Loops, Breaking Out Of Loops, Switch Statements, Functions; JavaScript Events, Selecting HTML elements using get Element ById(), **jQuery**: jQuery Events, jQuery Effects, jQuery HTML, jQuery Traversing,	5
V	**Unit V: Server Side Scripting** **PHP & Mysql**: Basic Syntax, defining variable and constant, PHP Data type, Operator and Expression, Handling Html Form with PHP: Capturing Form Data, dealing with Multi-value filed, redirecting a form after submission, PHP Session. Connection with MySQL Database, Performing basic database operation (DML)(Insert, Delete, Update, Select)	6
	Total	**26**

Textbook Books:

1. HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, Ajax, PHP and jQuery, 2ed (English, Paperback, DT Editorial Services)
2. Full-Stack JavaScript Development by Eric Bush.

Reference Books:

1. Robin Nixon, Learning PHP, MySQL & JavaScript with j Query, CSS & HTML5 Paperback by Orielly Pub.
2. E. Robson, E. Freeman, Head First HTML & CSS, O'Reilly Media, 2nd Edition, 2012.
3. Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step-by-Step Guide to Creating Dynamic Websites by Robin Nixon.
4. Web Design with HTML, CSS, JavaScript, and JQuery Set Book by Jon Duckett Professional JavaScript for Web Developers Book by Nicholas C. Zakas.

5. Head First jQuery by Ryan Benedetti, Ronan Cranley, September 2011, O'Reilly Media, Inc.

Digital Material: (NPTEL)

1. Introduction to Modern Application Development

Sr No.	Web Technology Lab: Practical Assignment
1.	Create a simple HTML page displaying personal details using text formatting tags.
2.	Design a web page that includes an image, hyperlink, and a nested table.
3.	Create a form with fields: name, email, gender, date of birth, and submit button.
4.	Apply internal, external, and inline CSS to style a web page with headings and tables.
5.	Create a responsive webpage using Bootstrap classes for typography, buttons, alerts, and tables. Apply proper layout and spacing
6.	Write a JavaScript program for a simple calculator using functions
7.	Write a JavaScript program to validate form inputs (e.g., email, empty fields)
8.	Develop and demonstrate a HTML file that includes JavaScript script that uses String functions
9.	Develop and demonstrate a HTML file that includes JavaScript for find all the prime numbers between starting and ending number.
10.	Create a dynamic webpage that responds to user actions like clicks, hover, and form input using jQuery event listeners and animation effects.
11.	Write a PHP program to capture form data and display it
12.	Write a PHP program to implement sign-In and Sign-out functionality.
13.	Write a PHP program to perform CRUD operation on the records which are stored in the database.

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: III	
Course:	Python Programming Lab				Code: T25CPP208	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
3	-	2	-	25	25	50

Prior knowledge of c or c++ programming

Course Objectives:

This course aims at enabling students:

1. Understand Python syntax, data types, operators, and control structures for problem solving.
2. Apply functions, modules, and built-in libraries to develop reusable programs.
3. Implement core data structures (list, tuple, set, dictionary) for efficient computation.
4. Use file handling and exception handling for robust real-world applications & Build object-oriented solutions using classes, objects, and inheritance.

Course Outcomes:

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

CO1: Explain Python syntax, data types, and operators to write basic programs.

CO2: Apply conditional statements and loops to solve computational problems.

CO3: Implement programs using Python strings and built-in functions/methods.

CO4: Use Python data structures (list/tuple/set/dictionary) for data organization and processing.

CO5: Develop modular programs using functions, modules, and exception handling & Build basic object-oriented solutions using classes, inheritance, and file handling.

Detailed Syllabus

Unit	Description	Duration (H)
I	Python features, applications, installation, IDEs,Python program structure, comments, indentation, Variables, input/output, type conversion ,Operators, expressions Built-in functions: type(), id(), len(), etc.	5
II	**Control Flow & Strings : Decision making:** if, elif, else, nested conditions ,Looping: for, while, loop control (break, continue, pass) **Strings:** indexing, slicing, immutability **String methods:** upper(), lower(), strip(), split(), replace(), find(), count()	5

III	**Python Data Structures** **Lists:** creation, indexing/slicing, methods (append, extend, insert, pop, remove, sort) **Tuples:** packing/unpacking, methods (count, index) **Sets:** set operations, use-cases **Dictionaries:** key-value operations, methods, iteration **Comprehensions:** list/dict comprehension	5
IV	**Functions, Modules & Packages: Functions:** definition, return, parameters (default, keyword, *args, **kwargs) **Scope:** local/global Lambda, map(), filter(), reduce() (intro) Modules: importing, creating modules Packages (intro), pip	5
V	**File handling:** open/close, read/write/append Text vs binary files, **Exceptions:** try-except-else-finally, multiple exceptions,User-defined exceptions, **Object Oriented Programming & Libraries** OOP concepts: class, object, constructor, methods Inheritance, polymorphism (basic), encapsulation	6
Total		**26**

Text Books:

1. Reema Thareja – Python Programming: Using Problem Solving Approach (Oxford University Press)
2. Yashavant Kanetkar – Let Us Python (BPB Publications)
3. R. Nageswara Rao – Core Python Programming (Dreamtech Press)

Reference Books:

1. **E. Balagurusamy** – *Programming with Python* (McGraw-Hill)
2. **Wes McKinney** – *Python for Data Analysis* (for NumPy/Pandas basics)
3. **Mark Lutz** – *Learning Python*
4. **Eric Matthes** – *Python Crash Course*

e-sources: NPTEL/SWAYAM Course (if any)

1. **Programming, Data Structures and Algorithms using Python (NPTEL)** — Prof. **Madhavan Mukund** (Chennai Mathematical Institute).
2. **The Joy of Computing using Python (NPTEL)** — Prof. **Sudarshan Iyengar** (IIT Ropar).
3. **Python 3.4.3 (SWAYAM – Spoken Tutorial, IIT Bombay)** — Prof. **Kannan Moudgalya.**

Shri Vile Parle Kelavani Mandal's

SVKM NMIMS Global University, Dhule

School of Technology, Management & Engineering

Master List of Practical

Sr. No.	PRACTICAL DESCRIPTION
1.	Write a Program to calculate area of triangle, rectangle, circle
2.	Write a Program to find the union of two lists.
3.	Write a Program to find the intersection of two lists.
4.	Write a Program to remove the —i^{th} occurrence of the given word in a list where words repeat.
5.	Write a Program to count the occurrences of each word in a given string sentence.
6.	Write a Program to check if a substring is present in a given string.
7.	Write a Program to map two lists into a dictionary.
8.	Write a Program to count the frequency of words appearing in a string using a dictionary.
9.	Write a Program to create a dictionary with key as first character and value as words starting with that character.
10.	Write a Program to find the length of a list using recursion.
11.	Write a Program to compute the diameter, circumference, and volume of a sphere using class
12.	Demonstrating Method Overriding through Polymorphism
13.	Write a Program to read a file and capitalize the first letter of every word in the file.

COMPUTER ENGINEERING

SEMESTER - IV
SYLLABUS

Program:	B. Tech. (Computer Engineering/DS/AIML)			Semester: IV		
Course:	Probability & Statistics			Code:T25CPS251		
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
4	3	-	1	40	60	100

Prior knowledge of

Course Objectives:

This course aims to enable students to:

1. **Understand and apply** the fundamental laws of probability to solve engineering problems.
2. **Apply** concepts of random variables and their probability distributions.
3. **Apply** standard theoretical probability distributions for data modeling and data analysis.
4. **Apply** correlation and regression techniques to study relationships between variables.
5. **Apply** statistical hypothesis testing methods for decision making.

Course Outcomes:

After successful completion of the course, the students will be able to:

CO1: Apply fundamental probability concepts, including axiomatic probability, addition and multiplication theorems, conditional probability, law of total probability, and Bayes' theorem, to solve engineering problems.

CO2: Apply concepts of discrete and continuous random variables, expectation, variance, probability density functions, cumulative distribution functions, moments, and moment generating functions.

CO3: Apply Binomial, Poisson, and Normal distributions to model and solve real-world data problems.

CO4: Apply correlation and regression techniques to determine relationships between variables.

CO5: Apply hypothesis testing methods such as t-test, z-test, and chi-square test for statistical decision making.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction to probability:** 1.1 Introduction to probability, Random experiment, Sample space, Events, Axiomatic Probability, 1.2 Addition theorem. 1.3 Conditional Probability, Multiplication theorem of Probability 1.4 Law of total Probability, Bayes theorem with proof	8
II	**Random variable and Expectation:** 2.1 Random Variables: Discrete & Continuous random Variables, 2.2 Expectation, Variance, Properties of Expectation & Variance 2.3 Probability Density Function, PMF & Cumulative Density Function. 2.3 Raw & Central Moments, Moment Generating Function.	7

III	**Theoretical probability distributions:** 3.1 Introduction 3.2 Binomial distribution, mean and variance of Binomial Distribution, fitting of Binomial distribution , Properties of Binomial distribution 3.3 Poisson distribution, mean and variance of Poisson Distribution fitting of Poisson distribution , Properties of Poisson distribution 3.4 Normal Distribution	8
IV	**Correlation and Regression:** 4.1 Correlation-Introduction & types of correlation, Karl Pearson's coefficient of correlation 4.2 Spearman's Rank correlation coefficient. 4.3 Relation between spearman's rank correlation coefficient (R) and Karl Pearson's coefficient of correlation(r) 4.4 Linear regression and two lines of regression. least square methods of linear regression	7
V	**Testing of hypothesis:** 5.1 Introduction, Hypothesis testing, Type I and Type II errors. 5.2 Tests of significance–Small Sampling Student's t- test, 5.3 Large sample test (z-test), 5.4 Non Parametric test: Chi-Square test goodness of fit.	9
Total		**39**

Text Books:

1. S.C. Gupta, Fundamentals of Statistics, Himalaya Publishing house, 7th edition.
2. G. V. Kumbhojkar, Probability and Random Processes, C. Jamnadas and Co., 14th Edition, 2010

Reference Books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. G. Haribaskaran, Probability, Queuing Theory and Reliability Engineering, Laxmi Publications, 2nd Edition, 2009.
3. Murray Spiegel, John Schiller, R. ALU Srinivasan, Probability and Statistics, Schaum's Outlines, 4th Edition, 2013.
4. Kishor S. Trivedi, Probability, Statistics with Reliability, Queuing and Computer Science Applications, Wiley India Pvt. Ltd, 2nd Edition, 2001.
5. Vijay K. Rohatgi, A. K. Md. Ehsanes Saleh, An Introduction to Probability And Statistics, Wiley Publication, 2nd Edition, 2001.
6. Roxy Peck, Chris Olsen, Jay Devore, Introduction to Statistics and Data Analysis, Third Edition, Thomson Books/Cole.
7. Ronald Walpole; Raymond Myers; Sharon Myers; Keying Ye, Probability & statistics for engineers & scientists, 9th edition, Prentice Hall

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: IV	
Course:	Probability & Statistics tutorial				Code: T25CPS251T	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
01	-	-	01	25	-	25
Minimum thirteen tutorials based on above syllabus will be conducted. Mini project relevant to the subject may be included, which would help the learner to apply the concept learnt.						

List of Suggested Tutorials

1. Addition theorem
2. Multiplication theorem of Probability
3. Bayes theorem
4. Expectation & Variance
5. Probability Density Function
6. Moments
7. Binomial distribution
8. Poisson distribution
9. Normal Distribution
10. Correlation
11. Regression
12. Least square methods of linear regression
13. Student's t- test
14. Z-test
15. Chi Square test

Program:	B. Tech. (Computer Engineering/DS/AIML)			Semester: IV		
Course:	Advanced Data Structure and Algorithms			Code: T25CDS252		
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
03	03	-	-	40	60	100

Prior knowledge of Data Structure

Course Objectives:

This course aims to enable students:

1. Understand fundamental algorithms, asymptotic analysis, and balanced tree and heap structures.
2. Use indexing, trees, and divide-and-conquer for efficient problem solving.
3. Use greedy, backtracking, and branch-and-bound for optimization and graph problems.
4. Design dynamic programming solutions and analyze problem complexity including NP-completeness.

Course Outcomes:

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

CO1 Apply algorithms, heaps, and balanced trees for efficient problem solving.

CO2 Apply indexing techniques, multiway trees, and divide-and-conquer methods to solve problems efficiently

CO3 Use greedy algorithms to efficiently solve optimization and graph problems.

CO4 Apply backtracking and branch and bound techniques to solve complex combinatorial and optimization problems efficiently.

CO5 Use dynamic programming to solve optimization problems and understand NP-completeness.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction to Algorithm, Balanced Trees and Heap:** Algorithm : Definition, Properties of Algorithms, Algorithm Design Techniques, Asymptotic Notations Order of Growth, Asymptotic Notations Recursion, Recurrences Relation, Substitution Method, Recursion Tree, Master Theorem. Heap and Balanced Trees : Basic concept of Heap,Max Heap and Min Heap using Binary Tree; Height balanced Tree: AVL Tree – insertion, deletion and searching on AVL Tree; Weight Balanced Tree	8
II	**Indexing and Multiway Trees, Divide and Conquer** **Indexing and Multiway Trees**: Indexing: Introduction, Indexing techniques; B-Tree, B+ Tree; Red Black Tree; Divide and Conquer-Binary Search, Quick Sort, Merge Sort, Strassen's	8

	Matrix Multiplication.	
III	**Greedy Algorithms** Introduction to Greedy Technique, Greedy Method, Optimal Merge Patterns, Huffman Coding, Knapsack Problem, Activity Selection Problem, Job Sequencing with Deadline, Single-Source Shortest Path Algorithm. Greedy algorithms for computing minimum spanning tree- Prims and Kruskal Algorithms.	8
IV	**Backtracking and Branch and Bound** Backtracking: Backtracking Concept, N–Queens Problem, Hamiltonian Cycle, Sum of Subsets Problem, Graph Coloring Problem, Branch and Bound: Introduction, Traveling Salesperson Problem, 15-Puzzle Problem, 0/1 Knapsack problem, Comparisons between Backtracking and Branch and Bound.	8
V	**Dynamic Programming** Concept, characteristics, overlapping subproblems and optimal substructure. DP design methodology including memoization and tabulation. 0/1 Knapsack, Matrix Chain Multiplication, Longest Common Subsequence, shortest paths: Bellman-Ford, Floyd Warshall. NP Completeness: Introduction, the Complexity Class P, the Complexity Class NP, Polynomial-Time Reduction, the Complexity Class NP-Complete.	7
	Total	**39**

Text Books:

1. S. Lipschutz, Data Structures, McGraw-Hill Publication, Revised 1st Edition, 2014.
2. Data structures and algorithms made easy, Career Monk Publications, 5th Edition

Reference Books:

1. Horowitz and Sahani, Fundamentals of Data Structures, Universities Press, 2nd Edition, 2008.
2. Thomas Cormen, Introduction to Algorithms, PHI Publication, 2nd Edition, 2002.

e-sources: NPTEL/SWAYAM Course (if any)

1. Design and Analysis of Algorithms, IIT Bombay Prof. Abhiram G Ranade, Prof. Ajit A Diwan, Prof. Sundar Viswanathan https://nptel.ac.in/courses/106101060

Program:		B. Tech. (Computer Engineering/DS/AIML)			Semester: IV		
Course:		Operating System			Code: T25COS253		
Credit		Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
		Lecture	Practical	Tutorial	ISE	ESE	Total
03		03	-	-	40	60	100

Prerequisites

- Data Structure and Computer Architecture & Organization

Course Objectives:

This course aims at enabling students:

1. To make students familiar with basic Concept of Operating System.
2. To Comprehend and Use concept of Process, Memory, File, and Disk Management of Operating System.

Course Outcomes:

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

1. To Comprehend and Use basic concepts of Operating System with its structure.
2. To Illustrate concepts of Process as well as Thread Management along with Implement concepts of CPU Scheduling algorithms.
3. To Illustrate concepts of Process Synchronization as well as deadlock along with Implement concepts of Synchronization primitives and banker's algorithms
4. To Comprehend concept of Memory Management along with Implement concepts of page replacement algorithms and memory allocation algorithms.
5. To Illustrate concepts of File System Manipulation as well as Disk Management along with Implement concepts of file allocation algorithms and disk secluding algorithms.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction and Operating System Structures:** Definition, Types of Operating system, Real-Time operating system, System Components: System Services, Systems Calls, System Programs, System structure, Virtual Machines, System Design and Implementation, System Generations.	7
II	**Processes and CPU Scheduling:** Process Concept, Process Scheduling, Operation on process, Inter-process Communication, Cooperating processes, Threads, Multithreading model, Scheduling criteria, Scheduling Algorithms, Thread Scheduling, Multiple-Processor Scheduling, Scheduling Algorithms evaluation.	8
III	**Process Synchronization:** The critical-section problem, Critical regions, Peterson's Solution, Synchronization Hardware, Semaphores, Classical Problems of synchronization, and Monitors **Deadlocks:** Systems Model, Deadlock characterization, Methods for	8

	handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock, Combined approach to deadlock Handling.	
IV	**Memory Management:** Basic concept, Logical and Physical address map, Memory allocation: Continuous Memory Allocation, Fixed and variable partition, Internal and external fragmentation and compaction, Paging: Principle of operation, Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging; Segmentation. **Virtual Memory:** Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page / Dirty bit – Demand paging, Page Replacement algorithms: Optimal, first in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used(LRU).	8
V	**File Management:** File Concept, Access methods, File types, File operation, Directory and disk structure, File System Structure, File System Implementation, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance. **Mass-Storage Structure:** Disk Structure, Disk attachment, Disk scheduling, Disk management, Swap Space Management.	8
	Total	**39**

Textbook Books:

1. Abraham Silberschatz, Peter B. Galvin and Greg Gagne, Operating System Concepts, Wiley Publication, 8th Edition, 2008.

Reference Books:

1. Andrew S. Tanenbaum, Modern Operating System, PHI Publication, 4th Edition, 2015.
2. D. M. Dhamdhere, Systems Programming and Operating Systems, McGraw-Hill, 2nd Edition, 1996. 3. Garry Nutt, Operating Systems Concepts, Pearson Publication, 3rd Edition, 2003.
3. Harvey M. Deitel, An Introduction to Operating Systems, Addison Wesley Publication, 2 nd Edition, 1990.
4. Thomas W. Doeppner, Operating System in Depth: Design and Programming, Wiley Publication, 2011.

Digital Material:

1. https://nptel.ac.in/courses/106105214
2. https://www.os-book.com/OS8/os8c/slide-dir/index.html

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: IV	
Course:	Entrepreneurship Development				Code: T25CEM254	
Credits	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
02	02	-	-	40	60	100

Pre-requisite: Effective Communication Skills

Course Objectives:

This course aims at enabling students:

1. To develop entrepreneurial mindset and skills among students.
2. To understand the process of starting and managing a new venture.
3. To familiarize students with business planning, financing, and legal aspects.
4. To enhance knowledge about innovation, startup ecosystem, and government support.
5. To build marketing, branding, risk management capabilities, and to build ethics and social responsibility in business contexts.

Course Outcomes:

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

CO1: Identify entrepreneurial opportunities and generate business ideas.

CO2: Analyze feasibility and prepare a detailed business plan.

CO3: Understand the financial, legal, and operational aspects of a startup.

CO4: Apply innovation and technology in developing entrepreneurial ventures.

CO5: Demonstrate managerial and leadership skills in running enterprises.

Detailed Syllabus

Module	Description	Duration (Hrs)
I	**Introduction to Value Education** Understanding Value Education, Self-exploration as the process for value education, Continuous happiness and prosperity as basic human aspirations, Right understanding, relationship, and physical facility, Happiness and prosperity: current scenario Methods to fulfill basic human aspirations	5
II	**Harmony in the Human Being** Human being as coexistence of the self and the body, needs of the self vs needs of the body, Body as an instrument of the self, Harmony in the self, Harmony of the self with the body, Programmed for self-regulation and health	5
III	**Harmony in the Family and Society** Family as the basic unit of human interaction, Values in human-to-human relationship, Trust as the foundational value, Respect as right evaluation,	5

	Harmony in society Vision of universal human order	
IV	**Harmony in Nature (Existence)** Understanding harmony in nature, Interconnectedness, self-regulation, and mutual fulfilment, Four orders of nature, Existence as coexistence at all levels, Holistic perception of harmony in existence	5
V	**Implications of Holistic Understanding – Professional Ethics** Natural acceptance of human values, Definitiveness of ethical human conduct, Humanistic education and universal human order, Competence in professional ethics, Holistic technologies and production systems, Management models and case studies, Transition strategies towards value-based life and profession	6
	Total	26

Textbooks:

1. Entrepreneurship Development and Small Business Enterprises. Poornima M. Charantimath. Pearson Education India

Reference books:

1. Entrepreneurship Development. S. Anil Kumar, S. C. Poornima, M. K. Abraham, K. Jayashree. New Age International Publishers
2. Fundamentals of Entrepreneurship. H. Nandan. PHI Learning
3. Business Model Generation. Alexander Osterwalder & Yves Pigneur. Wiley
4. Entrepreneurship in the 21st Century. A. C. Fernando. Pearson Education India

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: IV	
Course:	Basic Human Right				Code: T25CBR255	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
02	02	-	-	40	60	100

Pre-requisite: Knowledge of Constitution of India

Course Objectives:

This course aims at enabling students:

1. Understand the basic concepts of human rights, human values, justice, equality, and duties in society.
2. Develop awareness of the historical evolution of human rights, including key global movements and declarations.
3. Analyze the relationship between society, culture, religion, and human behavior in the context of rights.
4. Identify various human rights issues and violations, especially among vulnerable groups.
5. Understand the constitutional and legal framework of human rights in India.
6. Gain knowledge about national and international human rights institutions and mechanisms.

Course Outcomes:

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

1. To Explain the fundamental concepts and evolution of human rights and human values.
2. To Analyze the impact of social structures, culture, and religion on human rights and social issues.
3. To Identify and evaluate human rights violations affecting migrant workers and disabled individuals.
4. To Explain the provisions of human rights in the Indian Constitution, including fundamental rights and duties.
5. To Describe the role of national and international human rights organizations, such as NHRC, SHRC.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Fundamentals and Evolution of Human Rights** The Basic Concepts: individual, group, civil society, state, equality, justice, human values, human rights and human duties. Origin, Contribution of American bill of rights, French revolution, Declaration of independence, Rights of citizen, Rights of working and exploited people.	5

II	**Society, Culture, and Social Issues** Fundamental rights and economic programs, society, religion, culture, and their inter-relationship; impact of social structure on human behavior. **Social Structure and Social Problems:** Social and communal conflicts and social harmony, rural poverty, unemployment, bonded labor.	6
III	**Human Rights Issues and Vulnerable Groups** Migrant workers and human rights violations, human rights of mentally and physically challenged, state, individual liberty, freedom and democracy. **NGOs and human rights in India**: land, water, forest issues	5
IV	**Human Rights in the Indian Constitution** The Constitution of India: Preamble, Fundamental rights, Directive principles of state policy, Fundamental duties and some other provisions.	5
V	**Human Rights Framework and Institutions** Universal declaration of human rights and provisions of India, Constitution and law, National human rights commission and State Human Rights Commission.	5
Total		**26**

Text Books:

1. Shastry, T. S. N., India and Human rights: Reflections, Concept Publishing Company India (P Ltd.), 2005.

Reference Books:

1. Nirmal, C.J., Human Rights in India: Historical, Social and Political Perspectives (Law in India), Oxford India

e-sources: NPTEL

1. Human Rights in India By Prof. Rachana Kaushal

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: IV	
Course:	Modern Indian Languages Marathi (मराठी)				Code: T25CML256	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ICA	ESE	Total
01	1	0	0	25	0	25

पाठ्यक्रम उद्दिष्टे (Course Objectives):

1. CO1: विद्यार्थ्यांना आधुनिक मराठी भाषेचे व्याकरण, संवाद कौशल्य आणि साहित्यिक संकल्पना यांचे सखोल ज्ञान प्रदान करणे.
2. **CO2:** मराठी भाषेच्या सामाजिक, सांस्कृतिक आणि ऐतिहासिक विकासाचा अभ्यास करून भाषिक जाणिवा विकसित करणे.
3. **CO3:** शैक्षणिक, व्यावसायिक लेखन व भाषांतर कौशल्य विकसित करून विद्यार्थ्यांना व्यवहार्य भाषिक क्षमता प्रदान करणे.

अभ्यासक्रम परिणाम (Course Outcomes)

अभ्यासक्रम पूर्ण झाल्यानंतर विद्यार्थी:
CO1: आधुनिक मराठी भाषेचे व्याकरण शुद्धपणे वापरून प्रभावी संवाद साधू शकतील.
CO2: मराठी भाषेच्या विकासातील सामाजिक व सांस्कृतिक घटकांचे विश्लेषण करू शकतील.
CO3: शैक्षणिक, व्यावसायिक लेखन व मराठी-इंग्रजी भाषांतर प्रभावीपणे करू शकतील.

अभ्यासक्रम:

आठवडा	प्रयोगात्मक कार्य	तास
१.	मराठी लिपी आणि उच्चारांचा सराव, भाषेचा इतिहास स्वर, व्यंजन, संयुक्ताक्षरे यांचा अभ्यास शुद्ध उच्चार सराव मराठी भाषेचा उगम व विकास	1Hr
२.	साधी, संयुक्त व मिश्र वाक्यरचना; संवाद व अभिव्यक्ती वाक्यांचे प्रकार व रचना दिलेल्या वाक्यांचे रूपांतर दैनंदिन संभाषण सराव	1Hr
३.	इंग्रजी ते मराठी भाषांतर सोप्या वाक्यांवर आधारित भाषांतर शुद्ध व्याकरण व अर्थसुसंगतीवर भर	1Hr
४.	मराठी ते इंग्रजी भाषांतर लघुपरिच्छेदांचे भाषांतर शब्दसंग्रह व योग्य वाक्यरचना	1Hr

५.	औपचारिक पत्रलेखन रजा अर्ज विनंती पत्र तक्रार पत्र	1Hr
६.	अनौपचारिक पत्रलेखन मित्राला पत्र कुटुंबियांना पत्र निमंत्रण पत्र	1Hr
७.	परिच्छेद / सारांश लेखन (१००–१५० शब्द) दिलेल्या उताऱ्यावर आधारित सारांश सामाजिक विषयावर परिच्छेद	1Hr
८.	वृत्तलेखन कार्यक्रम अहवाल बातमी लेखन	1Hr
९.	संवाद लेखन परिस्थितीनुसार संवाद नाट्यसंवाद लेखन	1Hr
१०.	वाचन सराव शुद्ध उच्चार शब्दसंग्रह वाढ	1Hr
११.	संभाषण सराव मुलाखत सादरीकरण भूमिकानिर्मिती (Role Play)	2Hr
१२.	अंतिम प्रयोगात्मक परीक्षा व तोंडी परीक्षा **अंतर्गत परीक्षा (**Internal Test) **मौखिक परीक्षा** (Viva-Voce)	1Hr
	एकूण तास	13

पाठ्यपुस्तके:

मराठी भाषेचे मूळ – वि. का. राजवाडे
मराठी भाषेचा इतिहास – गं. बा. सरदार
मराठी भाषा: उगम व विकास – कृ. पा. कुलकर्णी
साहित्याची भाषा – डॉ. भालचंद्र नेमाडे

संदर्भ ग्रंथ:

सुगम मराठी व्याकरण व लेखन – मो. रा. वाळिंबे
अनुवाद: तत्त्वज्ञान आणि प्रयोग – डॉ. भालचंद्र नेमाडे
प्रशासकीय लेखन – भाषा संचालनालय महाराष्ट्र शासन
मराठी भाषा आणि साहित्याचा इतिहास – वि. का. राजवाडे
बनगरवाडी – लेखक: वि. पु. माडगूळकर
ढग – लेखिका: उषा शेळके

बटाट्याची चाळ – लेखक: पु. ल. देशपांडे

NPTEL / SWAYAM Digital Reference – SWAYAM

1. https://onlinecourses.swayam2.ac.in/nou25_lg30/preview

Program:	**B. Tech. (Computer Engineering/DS/AIML)**				**Semester: IV**	
Course:	**Operating Systems Laboratory**				**Code: T25COS257**	
Credit	**Teaching Scheme (Hrs./Week)**			**Evaluation Scheme and Marks**		
	Lecture	**Practical**	**Tutorial**	**ISE**	**ESE**	**Total**
1	**-**	**2**	**-**	**25**	**25**	**50**

Prerequisites:

- Structured Programming using C
- Data Structure and Computer Architecture & Organization

Course Objectives:

This course aims at enabling students:

1. To design and implement operating system principles such as system calls, CPU scheduling, process management, memory management, file systems, and deadlock resolution.
2. To develop and debug programs in Linux environment.

Course Outcomes:

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

CO1 Demonstrate basic Operating system Commands, Shell scripts, System Calls and API wrt Linux
CO2 Implement various process scheduling algorithms and evaluate their performance.
CO3 Implement and analyze concepts of synchronization and deadlocks.
CO4 Implement various Memory Management techniques and evaluate their performance.
CO5 Implement and analyze concepts of virtual memory.
CO6 Demonstrate and analyze concepts of file management and I/O management techniques.

Lis of Experiments

Sr. No.	**Description**
1	**Linux commands** Study basic Linux commands and implement any one basic commands of linux like ls, cp, mv and others using kernel APIs
2	**Shell Scripting** Write shell scripts to do the following: a. Display OS version, release number, kernel version b. Display top 10 processes in descending order c. Display processes with the highest memory usage. d. Display current logged in user and log name.
3	**Linux API** Create a child process in Linux using the fork system call. From the child process obtain the process ID of both child and parent by using getpid and getppid system

	call.
4	**CPU Scheduling:** Write a program to implement *CPU scheduling* algorithms - FCFS, SJF, and Priority
5	Write a program to implement *CPU scheduling* algorithms – Round Robin
6	**Process Synchronization:** Write a C program to implement solution of reader/writer problem or *Producer-Consumer problem* using semaphore.
7	**Deadlock:** Write a C program to implement *Banker's algorithm* for deadlock avoidance.
8	**Memory Management:** Write a C program to simulate *memory allocation algorithms* - First fit, Best fit and Worst fit.
9	**Virtual Memory:** Write a C program to implement *page replacement algorithms* such as first in first out, and least recently used page replacement.
10	**I/O Management:** Write a program in C to simulate *Disk Scheduling algorithms*- FCFS, SCAN, and SSTF.
11	**File Management:** Write a program to implement *file allocation strategy* as- continuous, linked and indexed.
Online Resources:	
1.	"BOSS Linux" on Spoken Tutorial platform.
2.	"BASH Shell Scripting" on Spoken Tutorial platform.

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: IV	
Course:	Data Science and Analytics Lab				Code: T25CDA258	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
01	-	02	-	25	25	50

Prior knowledge of

Course Objectives:

This course aims to enable students:

1. To develop data analytics code in Python
2. To be able to use python libraries for handling data
3. To develop analytical applications using python

Course Outcomes:

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

1. To Write python programs to handle data using Numpy and Pandas
2. To perform descriptive analytics
3. To Perform inferential data analytics
4. To Build models of predictive analytics

Laboratory Experiment List

Lab No	Description	Duration (H)
1	Write a Python program to perform following operations on NumPy array a) Creating a NumPy Array b) The Shape and Reshaping of NumPy Array c) 3. Expanding and Squeezing a NumPy Array	2
2	Write a Python program to perform following operations on NumPy array a) Indexing and Slicing of NumPy Array b) Stacking and Concatenating Numpy Arrays c) 3. Matrix Operations using NumPy 1d and 2d array	2
3.	Write a Python program to perform following operations on NumPy array a) Creating a DataFrame	2

	b) Concatenation and Setting Conditions c) Adding a new column d) Filling NaN with number and string e) Indexing and Slicing f) Sorting based on column values g) groupby()	
4	Write a Python program to read Text, CSV, Excel, Database, image and JSON file formats.	4
5	Write a Python program to demonstrate web scraping.	4
6	Write a Python program to compute Central Tendency, Dispersion and various Probability Distributions on the given Data.	2
7.	Write a Python program to perform Hypothesis Testing – One-tailed test and two-tailed test on the population mean on a given one sample data.	2
8.	Write a Python program to perform Hypothesis Testing using Z test ,T Test, ANOVA on a given two sample data.	4
9.	Write a Python program to perform a Chi-Square Test to test the independence of variables and find the goodness of fit of Poisson Distribution	2
Total		**26**

Text Books:

1. McKinney, W. (2012). Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. " O'Reilly Media, Inc.".
2. Douglas C. Montgomery, George C. Runger (2002). Applied Statistics & Probability for Engineering. "John Wiley & Sons, Inc"

Reference Books:

1. Swaroop, C. H. (2003). A Byte of Python. Python Tutorial.
2. Jay L. Devore (2011). Probability and Statistics for Engineering and the Sciences. "Cengage Learning".

e-sources: NPTEL/SWAYAM Course (if any)

1. https://onlinecourses.nptel.ac.in/noc24_cs20/

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: IV	
Course:	Community Engineering Project				Code: T25CEP259	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
01		02	-	50	-	50

Prior knowledge of

Course Objectives:

This course aims to enable students:

1. **To identify and analyze real-life community problems** by understanding their social, economic, and environmental contexts.
2. **To apply engineering knowledge and technical skills** to design practical, sustainable, and cost-effective solutions for community challenges.
3. **To work effectively in teams**, demonstrating leadership, collaboration, and project management skills.
4. **To communicate and interact with community stakeholders**, enhancing documentation, presentation, and interpersonal skills.
5. **To develop social responsibility, professional ethics, and innovative thinking** for contributing to community development and well-being.

Course Outcomes:

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

- CO1 To understand real-world community problems and identify appropriate engineering-based solutions addressing societal needs.
- CO2 To apply technical and interdisciplinary knowledge for designing sustainable, practical, and cost-effective solutions for community development.
- CO3 To demonstrate teamwork, leadership, and project management skills while working collaboratively on community-based projects.
- CO4 To communicate effectively with stakeholders through proper documentation, presentation, and interaction during project execution.
- CO5 To develop ethical responsibility, social awareness, environmental consciousness, and innovative thinking in solving community challenges using modern tools and technologies.

Detailed Syllabus

Unit	Description	Duration (H)

I	**Introduction to Community Engineering & Social Context** • Concept of community engineering and its importance • Role of engineers in society and sustainable development • Identification of community problems (rural/urban) • Basics of social, economic, and environmental issues • Case studies of successful community engineering projects	5
II	**Problem Identification & Need Analysis** • Techniques for problem identification (survey, field visits, interviews) • Stakeholder analysis and community participation • Data collection methods and tools • Need assessment and prioritization of problems • Feasibility study (technical, economic, social)	5
III	**Design & Planning of Solutions** • Conceptual design of engineering solutions • Sustainable and eco-friendly design approaches • Use of appropriate technology for community development • Resource planning (materials, cost, manpower) • Risk assessment and mitigation strategies	6
IV	**Implementation & Project Management** • Execution of community-based projects • Project management techniques (planning, scheduling, monitoring) • Teamwork, leadership, and coordination • Documentation of project activities • Ethical practices and professional responsibility	5
V	**Evaluation, Impact Analysis & Reporting** • Performance evaluation of the project • Social, economic, and environmental impact assessment • Feedback from stakeholders • Preparation of technical report and documentation • Presentation and demonstration of project outcomes	5
Total		**26**

Text Books:

1. A Systems Approach to Modeling Community Development Projects, Bernard Amadei,2018
2. Community Development Social Action and Social Planning A Practical Guide, Alan Twelvetrees, Russell Todd,2014

Reference Books:

1. Engineering Applications in Sustainable Design and Development, Bradley Striebig, 2008.
2. Engineering in Plain Sight, Grady Hillhouse, 2nd Edition, 2018.

e-sources: NPTEL/SWAYAM Course (if any)

1. **Sustainable Engineering**
   By Prof. Bhavik R. Bakshi, The Ohio State University (NPTEL)
   https://nptel.ac.in/courses/112/106/112106329
2. **Environmental Engineering and Sustainability**
   By Prof. Sudha Goel, IIT Kharagpur
   https://nptel.ac.in/courses/120/105/120105019
3. **Rural Development: Concepts and Practices**
   By Prof. V. K. Jain, IIT Roorkee
   https://nptel.ac.in/courses/109/107/109107169

Shri Vile Parle Kelavani Mandal's

SVKM NMIMS Global University, Dhule

School of Technology, Management & Engineering

Multidisciplinary Minor Courses offered by Department of Computer Engineering

Sr. No.	Semester	Course	Teaching Scheme				End Semester Examination						Continuous Assessment			Aggregate (A+B)	Category	Mode of Examination	Template
			Theory (hrs.)	Practical (hrs.)	Tutorial (hrs.)	Credits	Duration (hrs.)	Theory	Oral	Practical	Oral & Practical	ESE Total (A)	Term Test	Term Work	CAT otal (B)				
1	IV	Python Programming	2	-	-	2	2	60	-	-	-	60	40	-	40	100	Theory	Written	3604
2		Python Programming Lab	-	2	-	1	2	-	-	-	25	25	-	25	25	50	Lab	Lab	3606
3	V	Data Analytics	3	-	-	3	2	60	-	-	-	60	40	-	40	100	Theory	Written	3604
4	VI	Artificial Intelligence	3	-	-	3	2	60	-	-	-	60	40	-	40	100	Theory	Written	3604
5	VII	Prompt Engineering	2	-	-	2	2	60	-	-	-	60	40	-	40	100	Theory	Written	3604
6	VIII	Machine & Deep Learning	3	-	-	3	2	60	-	-	-	60	40	-	40	100	Theory	Written	3604

Multidisciplinary Minor Courses offered by Department of Computer Engineering /Computer Science and Engineering-Data Science/Computer Science and Engineering-Artificial Intelligence

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: IV	
Course:	Python Programming				Code: T25CPP260	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
02	02	-	-	40	60	100

Prior knowledge of

Course Objectives:

This course aims to enable students:

1. To introduce the fundamentals of Python programming, including syntax, data types, operators, and control flow constructs.
2. To develop problem-solving skills using decision control statements, loops, and built-in data structures in Python.
3. To enable students to design modular Python programs using functions, modules, and standard libraries while following good programming practices.
4. To familiarize students with string manipulation techniques and different programming paradigms, especially object-oriented programming concepts.
5. To provide knowledge of class design, object creation, and file handling mechanisms for real-world Python applications.

Course Outcomes:

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

1. Write and execute basic Python programs using variables, data types, operators, expressions, indentation rules, and input/output operations.
2. Apply conditional statements, loops, and built-in data structures (lists, tuples, dictionaries, and sets) to solve computational problems effectively.
3. Develop modular Python programs using user-defined functions, lambda functions, modules, packages, and standard library components with proper documentation.
4. Manipulate strings efficiently using built-in operations and methods, and differentiate programming paradigms with emphasis on object-oriented principles.
5. Design Python programs using classes and objects and perform file handling operations for persistent data storage.

Detailed Syllabus		
Unit	**Description**	**Duration (H)**
I	**Basics of Python Programming**: Features of Python, History and Future of Python, Writing and executing Python program, Literal constants, variables and identifiers, Data Types, Input operation, Comments, Reserved words, Indentation, Operators and expressions, Expressions in Python.	04 Hrs
II	**Decision Control Statements**: Decision control statements, Selection/conditional branching Statements: if, if-else, nested if, if-elif-else statements. Basic loop Structures/Iterative statements: while loop, for loop, selecting appropriate loop. Nested loops, break, continue, pass, else statement used with loops**. Data Structures and its Operations**- Tuples, Lists, Dictionary, and Set.	06 Hrs
III	**Functions and Modules:** Need for functions, Function: definition, call, variable scope and lifetime, return statement. Defining functions, Lambda or anonymous function, documentation string, good programming practices. Introduction to modules, Introduction to packages in Python, Introduction to standard library modules.	04 Hrs
IV	**Strings and Operations**- concatenation, appending, multiplication and slicing. Strings are immutable, strings formatting operator, built in string methods and functions. Slice operation, ord() and chr() functions, in and not in operators, comparing strings, Iterating strings, the string module.	04 Hrs
V	**Programming Paradigms**-Monolithic, Procedural, Structured and Object-oriented, Features of Object oriented programming **Classes and Objects**: classes and objects, class method and self object, class variables and object variables, public and private members, class methods. **File Handling**: Introduction, File path, Types of files, Opening and Closing files, Reading and Writing files.	08 Hrs
	Total	**26**

Text Books:

1. Michael Urban and Joel Murach, Murach's Python Programming, Murach's Publication, 2016.

Reference Books:

1. Charles Severance, Python for Informatics: Exploring Information, University of Michigan, Version 2.7.0, 2014.
2. Dr. R. Nageswara Rao, Core Python Programming, Dreamtech Press, 1st Edition, 2016.
3. Mark Lutz, Learning Python, O'Reilly Media, 5th Edition, 2013.
4. Mark Pilgrim, Dive into Python 3, A press Publication, 2nd Edition, 2009.
5. Allen B. Downey, Think Python, O'Reilly Media, 2nd Edition, 2012.Jon Kleinberg and Eva

Tardos, Algorithm Design, Pearson Education, 1st Edition, 2006.

e-sources: NPTEL/SWAYAM Course (if any)

2. https://onlinecourses.nptel.ac.in/noc24_cs57/preview
3. https://onlinecourses.nptel.ac.in/noc25_cs59/preview

Program:	B. Tech. (Computer Engineering/DS/AIML)			Semester: IV		
Course:	Python Programming Laboratory			Code: T25CPP260L		
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
01		02	-	25	25	50

Prior knowledge of

Course Objectives:

This course aims to enable students:

1. To introduce the fundamentals of Python programming, including syntax, data types, operators, and control flow constructs.
2. To develop problem-solving skills using decision control statements, loops, and built-in data structures in Python.
3. To enable students to design modular Python programs using functions, modules, and standard libraries while following good programming practices.
4. To familiarize students with string manipulation techniques and different programming paradigms, especially object-oriented programming concepts.
5. To provide knowledge of class design, object creation, and file handling mechanisms for real-world Python applications.

Course Outcomes:

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

1. Write and execute basic Python programs using variables, data types, operators, expressions, indentation rules, and input/output operations.
2. Apply conditional statements, loops, and built-in data structures (lists, tuples, dictionaries, and sets) to solve computational problems effectively.
3. Develop modular Python programs using user-defined functions, lambda functions, modules, packages, and standard library components with proper documentation.
4. Manipulate strings efficiently using built-in operations and methods, and differentiate programming paradigms with emphasis on object-oriented principles.
5. Design Python programs using classes and objects and perform file handling operations for persistent data storage.

Detailed Syllabus

Sr. No.	Description
1	Program to calculate the area of triangle, rectangle, circle.
2	Program to find the union of two lists.
3	Program to find the intersection of two lists.
4	Program to remove the i^{th} occurrence of the given word in a list where words repeat.
5	Program to count the occurrences of each word in a given string sentence.
6	Program to check if a substring is present in a given string.
7	Program to map two lists into a dictionary.

8	Program to count the frequency of words appearing in a string using a dictionary.
9	Program to create a dictionary with key as first character and value as words starting with that character.
10	Program to find the length of a list using recursion.
11	Write a Program to compute the diameter, circumference, and volume of a sphere using class.
12	Program to read a file and capitalize the first letter of every word in the file.

Text Books:

1. Michael Urban and Joel Murach, Murach's Python Programming, Murach's Publication, 2016.

Reference Books:

1. Charles Severance, Python for Informatics: Exploring Information, University of Michigan, Version 2.7.0, 2014.
2. Dr. R. Nageswara Rao, Core Python Programming, Dreamtech Press, 1st Edition, 2016.
3. Mark Lutz, Learning Python, O'Reilly Media, 5th Edition, 2013.
4. Mark Pilgrim, Dive into Python 3, A press Publication, 2nd Edition, 2009.
5. Allen B. Downey, Think Python, O'Reilly Media, 2nd Edition, 2012.Jon Kleinberg and Eva Tardos, Algorithm Design, Pearson Education, 1st Edition, 2006.

e-sources: NPTEL/SWAYAM Course (if any)

1. https://onlinecourses.nptel.ac.in/noc24_cs57/preview
2. https://onlinecourses.nptel.ac.in/noc25_cs59/preview

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: V	
Course:	Data Analytics				Code: T25CDA310	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
03	03	-	-	40	60	100

Prior knowledge of

Course Objectives:

This course aims at enabling students:

1. To provide a rigorous understanding of data quality and the principles of computational statistics, ensuring students can identify and mitigate sampling bias and data errors.
2. To familiarize students with descriptive measures and exploratory techniques required to summarize data distributions and select appropriate statistical models based on complexity.
3. To teach the mathematical logic behind various data transformations (Power, Log, and Logit) to prepare non-normal datasets for parametric analysis.
4. To instill a deep understanding of the classical hypothesis testing framework, including the trade-offs between Type I/II errors, significance levels, and statistical power.
5. To introduce advanced multi-group comparison techniques (ANOVA/MANOVA) and categorical association tests for complex experimental designs.

Course Outcomes:

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

CO1 Perform a comprehensive data audit to identify confounding variables, handle missing values, and minimize bias in datasets.

CO2 Compute higher-order moments (Skewness/Kurtosis) and use statistical indices to determine the most efficient model for a given data structure.

CO3 Apply Box-Cox and other power transformations to standardize skewed data, making it suitable for classical statistical testing.

CO4 Conduct and interpret Z-tests, T-tests, and Goodness-of-Fit tests (like Shapiro-Wilk) to draw statistically significant conclusions from samples.

CO5 implement ANOVA and Non-Parametric alternatives (like Kruskal-Wallis) to compare multiple populations and evaluate associations in contingency tables.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Data Integrity, Foundations, and Quality** **Statistical Concepts:** Differences between Statistical Methods and Computational Statistics. **Data Quality:** Data preparation and cleaning; identifying and handling missing data and data errors. **Sampling & Bias:** Sampling techniques, determining sample size, and Degrees of Freedom.	07

	Critical Evaluation: Understanding statistical error, bias, confounding variables, and the common misuse/misinterpretation of data. **Modeling Foundations:** Introduction to Statistical Modeling and Inference.	
II	**Descriptive Statistics and Model Selection** **Centrality & Spread:** Counts, specific values, Measures of Central Tendency (Mean, Median, Mode), and Measures of Spread (Variance, Standard Deviation). **Shape & Distribution:** Measure of distribution shape (Skewness, Kurtosis), Moments, and Key Statistical Functions. **Exploratory Data Analysis (EDA):** Visual tools for identifying patterns and outliers. **Model Assessment:** Statistical indices, measures of complexity, and criteria for model selection.	07
III	**Mathematical Transformations and Standardization** **Power Transformations:** Box-Cox and Freeman-Tukey (square root and arcsine) transforms. **Scaling Techniques:** Logarithmic and Exponential transforms. **Probability Mapping:** Logit transforms and Normal transforms for achieving Gaussian distributions. **Standardization:** Z-score normalization and its role in comparative analysis.	07
IV	**Classical Hypothesis Testing and Goodness of Fit** **Hypothesis Framework:** Null and Alternative hypotheses, Types of error (Type I and Type II), p-values, Statistical Significance, Power, and Robustness. **Goodness of Fit Tests:** Anderson-Darling, Chi-square, Kolmogorov-Smirnov, Ryan-Joiner, Shapiro-Wilk, Jarque-Bera, and Lilliefors. **Z-Tests:** Single mean (known variance), difference between two means, and tests for proportions. **T-Tests:** Single mean (unknown variance), difference between two means, and testing of regression coefficients. **Confidence Intervals:** Estimating population parameters with specific margins of error.	07
V	**Analysis of Variance (ANOVA) and Contingency Analysis** **Variance & Homogeneity:** Chi-square test of single variable, F-test of two variables, and tests of homogeneity. **Parametric ANOVA:** One-way and Two-way ANOVA, MANOVA (Multivariate), and ANCOVA (Analysis of Covariance). **Non-Parametric Analysis:** **Group Comparison:** Kruskal-Wallis, Friedman ANOVA, and Mood's Median test.	07

	Rank Tests: Wilcoxon rank-sum/Mann-Whitney U test and the Sign test. **Contingency Tables:** Chi-square contingency test, G-test, Fisher's Exact test, McNemar's test, and various Measures of Association.	
Total		**35**

Text Books:

1. Dr. Michael J de Smith, Statistical Analysis Handbook: A Comprehensive Guide to Statistical Concepts, Methods and Tools, The Winchelsea Press, Drumlin Security Ltd, Edinburgh, 2018 edition. https://www.statsref.com/HTML/index.html

Reference Books:

1. Douglas C. Montgomery, George C. Runger, Applied Statistics and Probability for Engineers, Sixth Edition, Wiley, 2013.
2. Dr.J.Ravichandran, Probability And Statistics For Engineers, First Edition, Wiley, 2010.
3. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman. Mining of Massive Datasets. V2.1, Cambridge University Press. 2014. (free online)
4. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. ISBN 0262018020, 2013.
5. "Applied Statistics and Probability for Engineers" – Douglas C. Montgomery.
6. "Practical Statistics for Data Scientists" – Peter Bruce & Andrew Bruce.
7. "Statistical Inference" – George Casella and Roger L. Berger.
8. Foster Provost and Tom Fawcett. Data Science for Business: What You Need to Know about Data Mining and Data-analytic Thinking. ISBN 1449361323. 2013.
9. Cathy O'Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline, O'Reilly. 2014.

e-sources: NPTEL/SWAYAM Course (if any)

1. Swayam - Data Analytics with Python, by Prof. A. Ramesh (IIT Roorkee).
2. Data Science for Engineers by Prof. Raghunathan Rengasamy (IIT Madras).

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: VI	
Course:	Artificial Intelligence				Code: T25CAI360	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
03	03	-	-	40	60	100

Prior knowledge of

Course Objectives:

This course aims at enabling students:

1. To introduce the fundamental concepts, principles, and methodologies of Artificial Intelligence (AI) and Computational Intelligence (CI).
2. To develop the ability to analyze, design, and implement AI algorithms using heuristic and search-based approaches.
3. To understand and evaluate various knowledge representation, reasoning, and planning mechanisms in intelligent systems.
4. To apply computational intelligence techniques such as neural networks, fuzzy logic, for real-world problem solving.

Course Outcomes:

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

CO1 Explain the foundational concepts, historical evolution, and problem-solving paradigms of Artificial and Computational Intelligence.

CO2 Design and implement AI algorithms using heuristic and search-based methods.

CO3 Utilize computational intelligence methods such as neural networks, fuzzy logic, a solve engineering problems.

CO4 Analyze and evaluate real-world problems using appropriate AI and CI approaches, including knowledge representation and probabilistic reasoning.

CO5 Demonstrate research orientation and critical understanding of modern AI topics such as NLP, learning, and planning systems.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Foundations of Artificial Intelligence:** Introduction to Artificial Intelligence, History, and Philosophy of AI. Intelligent agents: structure, environment, types of agents.AI Problem formulation, State-space representation, Problem-solving frameworks. The Concept of Rationality	8
II	**Problem-solving:** Solving Problems by Searching, Problem-Solving Agents, Uninformed search: Depth-first, Breadth-first, Uniform cost search. Informed (Heuristic) search: Greedy search, A* search, AO* algorithm, Local search algorithms: Hill Climbing, Constraint Satisfaction Problems (CSPs), Backtracking algorithms, heuristics for CSPs. Game Playing: Minimax Algorithm, Alpha-Beta Pruning.	8

III	**Knowledge Representation and Reasoning:** Knowledge representation issues, Representation & mapping, Approaches to knowledge representation, Issues in knowledge representation. Using predicate logic: Representing simple fact in logic, Representing instant & ISA relationship, Computable functions & predicates, Procedural verses declarative knowledge, Logic programming, Forward verses backward reasoning	8
IV	**Probabilistic Reasoning:** Representing knowledge in an uncertain domain, The semantics of Bayesian networks, Dempster-Shafer theory, fuzzy logics, Planning: Overview, Components of a planning system, Goal stack planning, Hierarchical planning and other planning techniques.	8
V	**Natural Language Processing:** Introduction, Syntactic processing, Semantic analysis, Discourse & pragmatic processing. Learning: Forms of learning, Inductive learning, Learning decision trees, explanation based learning, Learning using relevance information, Neural net learning & genetic learning.	8
Total		**40**

Text Books:

1. Stuart Russell, Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, Pearson Education, 2020.
2. Rich, E. and Knight K.: Artificial Intelligence, Tata McGraw- Hill
3. Andries Engelbrecht, Computational Intelligence: An Introduction, 2nd Edition, Wiley, 2007.

Reference Books:

1. Kevin Warwick, Artificial Intelligence Techniques in Power Systems, IET Publications.
2. Timothy J. Ross, Fuzzy Logic with Engineering Applications, 4th Edition, Wiley, 2016.
3. Melanie Mitchell, An Introduction to Genetic Algorithms, MIT Press, 1998.

e-sources: NPTEL/SWAYAM Course (if any)

1. https://onlinecourses.nptel.ac.in/noc25_cs07/

2. https://onlinecourses.swayam2.ac.in/nou25_cs07

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: VII	
Course:	Prompt Engineering				Code: T25CPE410	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
02	02	-	-	40	60	100

Prerequisite: Python Programming, AI

Course Objectives:

This course aims to enable students:

1. Explain the evolution of NLP to large language models, their architecture, and the role of prompt engineering in AI/ML applications.
2. Design and evaluate effective prompts using various prompting types, patterns, and debugging techniques.
3. Apply advanced prompting methods, including Chain-of-Thought, ReAct, and multimodal prompting, using tools such as LangChain and OpenAI SDK.
4. Develop domain-specific prompts for tasks such as coding assistance, conversational agents, and data analysis, including low-resource language applications.
5. Assess prompt safety, bias, and ethical considerations, and implement best practices for secure and responsible LLM usage.

Course Outcomes:

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

1. Explain the evolution of Natural Language Processing to Large Language Models, describe the architecture of LLMs, and illustrate the significance of prompt engineering in AI/ML systems.
2. Design, analyze, and evaluate effective prompts using various prompting techniques, patterns, and debugging strategies based on standard performance and safety metrics.
3. Apply advanced prompting techniques and software frameworks to develop domain-specific prompt solutions for tasks such as coding, conversational systems, data analysis, and low-resource language processing.
4. Assess ethical, safety, and bias-related issues in prompt engineering and implement best practices for secure, responsible, and robust deployment of Large Language Models.
5. Analyze existing prompts and make strategic combinations for enhanced prompts.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction to Prompt Engineering & LLMs**: Evolution of NLP to Large Language Models, **Anatomy of LLMs:** Tokens, embeddings, context windows, Why prompt engineering matters, Prompt engineering applications in AI/ML systems, Introduction to GPT, Claude, PaLM, Mistral, etc., Prompting vs fine-tuning.	05 Hrs

II	**Foundations of Prompt Design:** Types of prompting: Zero-shot, One-shot, Few-shot, Prompt patterns: Instructional, Delimiting, Role-based, Socratic, Prompt evaluation metrics: accuracy, relevance, hallucination, safety, Prompt debugging techniques. **Advanced Prompting Techniques:** Chain-of-Thought Prompting, Self-Ask Prompting, ReAct (Reason + Act) prompting, Multimodal prompting (text + image), Prompt templates using Python APIs (LangChain, OpenAI SDK).	06 Hrs
III	**Prompt Engineering for Domain-Specific Tasks:** Coding assistance and code generation prompts, Conversational agents and dialogue design, Legal, educational, medical prompts, Data wrangling and analysis via prompting, Prompting in low-resource languages.	04 Hrs
IV	**Prompt Safety, Bias, and Ethics:** Bias, toxicity, misinformation in model outputs, Adversarial prompting and jailbreak attempts, Safety best practices in prompt design, Evaluation frameworks (BLEU, ROUGE, BERTScore, human evals), Future directions: function calling, tool use, retrieval-augmented generation (RAG)	05 Hrs
V	**Introduction to Diffusion Models for Image Generation** Introduction to Image Generation with AI, Principles of Designing Prompts for Image Generation, Available Models - OpenAI DALL-E, Midjourney, Stable Diffusion, Google Gemini, Text to Video, Model Comparison, Reverse Engineering Prompts, Negative Prompts, Prompt Re-Writing, Prompt Analysis	06 Hrs
	Total	**26**

Text Books:

1. "The Art of Prompt Engineering with OpenAI APIs" by Nathan Hunter.
2. "Building AI Applications with OpenAI API" by Yash Sheth.

Reference Books:

1. AI Prompt Engineering Absolute Beginners Guide — Michael Miller
2. Prompt Engineering – Basics: Building Foundations for Effective AI Guidance — Raj Vaidyamath
3. Prompt Engineering for All: Training Handbook — Alamgir Hossain
4. Prompt Engineering for LLMs: The Art and Science of Building Large Language Model-Based Applications — John Berryman & Albert Ziegler

e-sources: NPTEL/SWAYAM Course (if any)

1. "Prompt Engineering for Everyone" by Isa Fulford & Andrew Ng DeepLearning.AI short course).
2. "Prompt Engineering for Generative AI", Google, https://developers.google.com/machine-learning/resources/prompt-eng

Program:	B. Tech. (Computer Engineering/DS/AIML)				Semester: VIII	
Course:	Machine & Deep Learning				Code: T25CML460	
Credit	Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks		
	Lecture	Practical	Tutorial	ISE	ESE	Total
03	03	-	-	40	60	100

Prior knowledge of

Course Objectives:

This course aims to enable students:

1. Understand the fundamentals of Machine Learning, its types, applications across industries, and perform essential data preprocessing using Python-based ML libraries.
2. Develop and implement core supervised learning algorithms such as regression, classification, and instance-based learning for solving real-world problems.
3. Evaluate machine learning models using appropriate performance metrics and apply model tuning techniques to improve generalization and reduce overfitting.
4. Analyze unlabeled data using unsupervised learning techniques including clustering and dimensionality reduction for pattern discovery and feature extraction.
5. Comprehend the foundations of deep learning and design neural network models including CNNs, RNNs, and GANs for complex data representations.

Course Outcomes:

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

CO1 Explain machine learning concepts, types, and applications, and perform data cleaning, transformation, and visualization using Python libraries such as NumPy, Pandas, and Matplotlib.

CO2 Implement supervised learning algorithms including Linear Regression, Logistic Regression, SVM, Decision Trees, Random Forests, KNN, and Naive Bayes for classification and regression tasks.

CO3 Assess machine learning models using suitable evaluation metrics and apply cross-validation and hyperparameter tuning techniques to address bias–variance tradeoffs.

CO4 Apply clustering and dimensionality reduction techniques such as K-Means, Hierarchical Clustering, DBSCAN, PCA, and t-SNE to discover hidden structures in data.

CO5 Design and analyze deep learning architectures including MLPs, CNNs, RNNs (LSTMs/GRUs), and GANs using appropriate activation functions and optimization techniques..

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction to Machine Learning**: What is Machine Learning, Overview of ML and its applications in various industries. Types of ML: Supervised Learning, Unsupervised Learning, Semi-supervised Learning, and Reinforcement	8

	Learning. **Data Preprocessing**: Data cleaning and preparation, Handling missing data, scaling, encoding categorical variables. **Introduction to Python for Machine Learning** Libraries: NumPy, Pandas, Matplotlib, Scikit-learn, TensorFlow, and Keras.	
II	**Supervised Learning Algorithms** **Linear Regression**: Simple Linear Regression and Multiple Linear Regression. Model evaluation using Mean Squared Error (MSE). **Logistic Regression**: Binary and Multinomial Logistic Regression. Understanding of Sigmoid function, odds, and probability. **Support Vector Machines (SVM):** Linear and non-linear classification using kernels. Margin maximization, soft margin, and regularization. **Decision Trees and Random Forests**: Decision tree construction (ID3, CART algorithms). Overfitting and pruning. Random Forests and Bagging techniques. **K-Nearest Neighbors (KNN)**: Instance-based learning, Distance metrics (Euclidean, Manhattan), KNN for classification and regression. **Naive Bayes Classifier:** Bayes' Theorem and its application to classification, Assumptions in Naive Bayes (independence of features).	8
III	**Model Evaluation and Tuning** **Cross-Validation**: K-fold cross-validation, Bias-Variance tradeoff. **Metrics for Classification**: Accuracy, Precision, Recall, F1-score, ROC-AUC curve. **Metrics for Regression**: Mean Absolute Error (MAE), Mean Squared Error (MSE), R^2 (Coefficient of Determination). **Hyperparameter Tuning**: Grid Search and Random Search for hyperparameter optimization. **Overfitting and Underfitting:** Techniques to avoid overfitting (regularization, dropout, cross-validation). **Unsupervised Learning Algorithms** **Clustering Algorithms**: K-Means Clustering, Hierarchical Clustering, DBSCAN, Understanding centroid-based, density-based, and agglomerative methods. **Dimensionality Reduction:** Principal Component Analysis (PCA), t-SNE (t-Distributed Stochastic Neighbor Embedding), Feature selection techniques.	8
IV	**Introduction to Deep Learning:** History of Deep Learning, Deep Learning Success Stories, McCulloch Pitts Neuron, Thresholding Logic, Perceptrons, Perceptron Learning Algorithm, Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Gradient Descent, Feed forward Neural Networks.	8
V	Deep Neural Networks, Activation functions, and Gradient Descent. Convolutional Neural Networks (CNNs) for image classification. Recurrent Neural Networks (RNNs) for	8

	sequential data (LSTMs, GRUs). Generative Adversarial Networks (GANs).	
Total		**40**

Text Books:

1. Machine Learning. Tom Mitchell. First Edition, McGraw- Hill, 1997.
2. Alpaydin, Ethem. Introduction to machine learning. MIT press, 2020.
3. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning, MIT Press.
4. Francois Chollet, Deep Learning with Python, Manning Publications.

Reference Books:

1. K.P. Soman, R. Longonathan and V. Vijay, Machine Learning with SVM and Other Kernel Methods, PHI-2009
2. Simon Haykin, Neural Networks and Learning Machines, Pearson Education.
3. Michael Nielsen, Neural Networks and Deep Learning, Free online book.
4. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer 2006.
5. Shalev-Shwartz,S., Ben-David,S., (2014), Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press
6. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, 2012.
7. Kishan Mehrotra, Chilukuri Mohan and Sanjay Ranka, Elements of Artificial Neural Networks, Penram International
8. Rajjan Shinghal, Pattern Recognition, Techniques and Applications, OXFORD

e-sources: NPTEL/SWAYAM Course (if any)

1. Introduction to Machine Learning, By Prof. Balaraman Ravindran , IIT Madras https://nptel.ac.in/courses/106106139
2. Introduction to Machine Learning, by Prof. S. Sarkar, IIT Kharagpur https://nptel.ac.in/courses/106105152
3. https://onlinecourses.nptel.ac.in/noc25_ee16
4. https://onlinecourses.nptel.ac.in/noc25_cs21/

B. Tech. Computer Engineering and Allied Branches Degree with Honours (Programme core additional 18 credits)

- B. Tech in Computer Engineering/ Computer Science and Engineering (Data Science)/ Computer Science and Engineering (AIML) with minor in other selected discipline in Engineering (as MDM) with Honours in "Cyber Security" under the Honours course from AY 2026-2027
- The criteria for awarding an Honours in "Cyber Security" to students are as follows:

  The student must secure 18 credits by completing the NPTEL Domain Certification in Cyber Security (https://nptel.ac.in/domains/discipline/106?course=106_5) during the B. Tech programme period from the second year to the final year of their graduation.
- This provision allows students to complete six or more additional courses (totalling 18 credits) in the discipline of Cyber Security, distributed across Semesters III to VIII, in addition to the prescribed 172 credits of the B.Tech. programme.

➢ **Eligibility Criteria to opt for Honours in Cyber Security**

Student must have CGPA equal to or greater than 7.5 at the end of second semester to go for this option.

Domain: Cyber Security

Core Courses

You have to complete 2 core courses

Please choose one course against each serial number for completing a core where there are multiple courses in a serial number.

SI No.	NPTEL Course ID	Course Name	SME Name	Institute	Course Duration	Credits
1	106105162	Cryptography and Network Security	Prof. Sourav Mukhopadhyay	IIT Kharagpur	12 weeks	3
2	106106144	Introduction to Operating Systems	Prof. Chester Rebeiro	IIT Madras	8 weeks	2
	106102132	**OR** Operating System	Prog. Sorav Bansal	IIT Delhi	12 weeks	3

	106105214	**OR** Operating System Fundamentals	Prof. Santanu Chattopadhyay	IIT Kharagpur	12 weeks	3

Elective Courses
You have to complete minimum 4 elective courses

SI No.	NPTEL Course ID	Course Name	SME Name	Institute	Course Duration	Credits
1	106105183	Computer Networks And Internet Protocol	Prof. Soumya Kanti Ghosh Prof. Sandip Chakraborty	IIT Kharagpur	12 weeks	3
2	106105194	Hardware Security	Prof. Debdeep Mukhopadhyay	IIT Kharagpur	12 weeks	3
3	106105217	Ethical Hacking	Prof. Indranil Sengupta	IIT Kharagpur	12 weeks	3
4	106106146	Privacy and Security in Online Social Media	Prof. Ponnurangam Kumaraguru	IIIT Hyderabad	12 weeks	3
5	106106199	Information Security-5-Secure Systems Engineering	Prof. Chester Rebeiro	IIT Madras	8 weeks	2
6	106106221	Foundations of Cryptography, IIIT Bangalore	Prof. Ashish Choudhury	IISc Bangalore	12 weeks	3
7	106108229	Secure Computation: Part I	Prof. Ashish Choudhury	IISc Bangalore	12 weeks	3
8	106106234	Systems and Usable Security	Prof. Neminath Hubballi	IIT Indore	4 weeks	1
9	106106235	Online Privacy	Prof. Ponnurangam Kumaragpur	IITH	12 weeks	3
10	106105235	Blockchain and its Applications	Prof. Sandip Chakraborty Prof. Shamik Sural	IIT Kharagpur	12 weeks	3
11	106108237	Secure Computation: Part II	Prof. Ashish Choudhury	IIIT Bangalore	12 weeks	3
12		Quantum	Prof. Shweta	IIT	12 weeks	3

	106106241	Algorithms and Cryptography	Agrawal	Madras		
13	106106248	Cyber Security and Privacy	Prof. Saji K Mathew	IIT Madras	12 weeks	3
14	106104467	Practical Cyber Security for Cyber Security Practitioners	Prof. Sandeep K. Shukla	IIT Kanpur	12 weeks	3



Dr. Makarand Shahade
Chairperson, BOS

Copy to: Office of Dean, STME