GATES Institute of Technology

(Autonomous)

GOOTY–515401(A.P)INDIA

==========

B.Tech(Regular-Full time)

(Effective for the students admitted into I Year from the Academic Year **2026-27**onwards)

Computer Science and Engineering

I Year Course Structure and Syllabus

And

B.Tech (Lateral Entry Scheme)

(Effective for the students getting admitted into II year through Lateral Entry Scheme from the Academic Year 2027-28 onwards)

B.Tech. –I Year I Semester

S.No.	Coursecode	Title	L	T	Pr	P	Credits
1	26BST08	Applied Chemistry for Computer Science	3	0	0	0	3
2	26BST02	Linear Algebra & Calculus	3	0	2	0	4
3	26MET02	Design Thinking	1	0	4	0	3
4	26CST01	Computational Thinking & Problem Solving with Python	3	0	0	0	3
5	26BST01	Communicative English	2	0	0	0	2
6	26BSP06	Applied Chemistry For Computer Science Lab.	0	0	0	3	1.5
7	26CSP01	Computational Thinking and Problem Solving with Python Lab	0	0	0	3	1.5
8	26BSP12	Health and wellness, Yoga and Sports	0	0	0	1	0.5
9	26BSP01	Communicative English Lab	0	0	0	2	1
Total			**12**	**0**	**6**	**9**	**19.5**

I Year II Semester

S.No.	Course code	Title	L	T	Pr	P	Credits
1	26AIP01	AI Tools and Applications	1	0	4	0	3
2	26BST13	Physics for Advanced Computing	3	0	0	0	3
3	26BST03	Differential Equations & Vector Calculus	3	0	2	0	4
4	26CST02	Data Structures &Efficient Problem Solving	2	0	2	0	3
5	26MET01	Engineering Graphics	1	0	4	0	3
6	26CSP02	Data Structures &Efficient Problem Solving Lab	0	0	0	2	1
7	26BSP11	Physics for Advanced Computing Lab	0	0	0	3	1.5
8	26BSP13	NSS/NCC/Scouts & Guides/Community Service	0	0	0	1	0.5
9	26MEP01	Engineering Workshop	0	0	0	3	1.5
10	26AIA01	Cyber security Awareness for Engineers	2	0	0	0	0
Total			**11**	**0**	**12**	**9**	**20.5**

I B.Tech I Semester

Course Code	APPLIED CHEMISTRY FOR COMPUTER SCIENCE	L	T	Pr	P	Credits
26BBST08		3	0	0	0	3
Prerequisite:	Basic Chemistry (Higher Secondary level)					

Course Objectives

- Understand the fundamentals of quantum chemistry including wave-particle duality, quantum numbers, and molecular orbital theory relevant to computing materials.
- Comprehend the chemistry and electronic properties of semiconductor materials and the chemical processes involved in semiconductor device fabrication.
- Evaluate electrochemical principles governing batteries, supercapacitors, and fuel cells as energy storage technologies for computing systems.
- Analyze corrosion mechanisms in electronic devices and design appropriate protective techniques for PCBs, solder joints, and interconnects.
- Explore the chemistry, synthesis methods, and properties of nanomaterials used in nanoelectronics, quantum computing, and sustainable electronics.

Course Outcomes (COs)

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

CO	Course Outcome Statement	Bloom 's Level
CO1	Apply quantum chemistry principles to analyze molecular systems and explain molecular orbital theory for computing materials.	L3–L4
CO2	Analyze semiconductor materials, doping chemistry, and fabrication processes used in modern electronic devices.	L3–L4
CO3	Evaluate energy storage technologies including batteries, supercapacitors, and fuel cells for computing and data centre applications.	L3–L4
CO4	Analyze corrosion mechanisms in electronic devices and recommend appropriate protective techniques for PCBs and interconnects.	L3–L4
CO5	Design advanced materials solutions using nanotechnology principles and AI-assisted tools for nanoelectronics and quantum computing.	L5–L6

UNIT – I	QUANTUM CHEMISTRY	9 Hrs

Content	Fundamentals of quantum chemistry: black body radiation, photoelectric effect, wave-particle duality, de Broglie wave equation, Heisenberg's uncertainty principle - fundamentals of quantum mechanics: Schrödinger wave equation; significance of Ψ and Ψ^2; quantum numbers - particle in a one-dimensional box - molecular orbital theory: bonding in homonuclear and heteronuclear diatomic molecules; energy level diagrams of O_2 and CO - π-molecular orbitals of

	butadiene and benzene; calculation of bond order.
AI Integration	Simulation of quantum systems using Python (NumPy, SciPy); visualization of molecular orbitals and energy levels using computational chemistry tools (Google Colab); introduction to quantum computing simulations using Qiskit (introductory level). Free/Open-Source AI Tools: Python (NumPy, SciPy), Google Colab, Qiskit.

UNIT – II	**CHEMISTRY OF SEMICONDUCTORS**	**9 Hrs**

Content	Atomic structure and chemical bonding relevant to semiconductors - band theory: valence band, conduction band, band gap energy - intrinsic and extrinsic semiconductors: silicon, germanium, gallium arsenide - doping chemistry: n-type (phosphorus, arsenic) and p-type (boron, aluminum) dopants - chemical vapour deposition (CVD) for thin film formation; silicon dioxide (SiO_2) formation and properties - chemistry of compound semiconductors: GaAs, InP, SiC.
AI Integration	AI-based prediction of semiconductor properties using ML models; process parameter optimization in semiconductor fabrication using machine learning; Python-based band gap prediction using regression models. Free/Open-Source AI Tools: Python, TensorFlow, Scilab.

UNIT – III	**ELECTRO CHEMISTRY AND ENERGY STORAGE**	**9 Hrs**

Content	Electrochemical cells: galvanic cells, electrode potential, Nernst equation and problems; potentiometry (redox reactions); conductometry (strong acid vs. strong base, weak acid vs. strong base) - primary batteries: zinc-air battery, sodium-air battery - secondary batteries: chemistry of charging and discharging of lead-acid, nickel-cadmium, nickel-metal hydride - lithium-ion batteries: construction, working, and applications of $LiCoO_2$ - supercapacitors: definition, classification, and applications - fuel cells: construction, working, and applications of hydrogen fuel cells and direct methanol fuel cells (DMFC) - battery management systems: overcharge protection and thermal management.
AI Integration	AI models for battery health prediction and state-of-charge optimization using scikit-learn; data analysis for energy efficiency in data centres using Python (Pandas); ML-based prediction of electrochemical impedance spectra. Free/Open-Source AI Tools: Python (Pandas, scikit-learn), Google Colab.

UNIT – IV	**CHEMISTRY OF CORROSION AND PROTECTIVE TECHNIQUES**	**9 Hrs**

Content	Corrosion: definition, examples, types — chemical corrosion and electrochemical corrosion; Pilling-Bedworth rule - Galvanic corrosion; concentration cell corrosion; factors affecting corrosion - Deal-Grove model: thermal oxidation of silicon; passivation layers — silicon dioxide, silicon nitride, and aluminium oxide - Corrosion of printed circuit boards (PCBs): copper oxidation, tin whiskers; corrosion in solder joints and interconnects - Protective techniques: proper design, using pure metals, metal alloys, inhibitors; cathodic protection — sacrificial anodic protection, impressed current cathodic protection - electroplating (nickel)

	and electro less plating (copper).
AI Integration	Computer vision (Open CV) for automated defect detection in PCBs; AI-based prediction of corrosion failure in electronic components using Tensor Flow; image classification models for identifying corrosion types from microscope images. Free/Open-Source AI Tools: Open CV, Tensor Flow, Python.

UNIT – V	CHEMISTRY OF NANOMATERIALS	9 Hrs

Content	Introduction to nanochemistry: size-dependent properties, quantum confinement - carbon nanomaterials: fullerenes, carbon nanotubes (CNTs), graphene - Synthesis methods: chemical vapour deposition (CVD), sol-gel process - quantum dots: synthesis, properties, applications in displays and quantum computing - metallic nanoparticles: preparation and applications of gold, silver, and copper nanoparticles - applications: nano electronics, interconnects, sensors (optical and biosensors), memory devices - spintronics materials: magnetic nanoparticles, topological insulators - chemistry of quantum computing materials: superconducting materials (NbTi, Nb_3Sn); ion trap materials.
AI Integration	AI-based nanomaterial property prediction using machine learning (Python, TensorFlow); simulation of nano-devices and quantum materials using open-source tools; generative AI for exploring nanomaterial design spaces. Free/Open-Source AI Tools: Python, TensorFlow, open-source materials simulation platforms.

Bloom's Taxonomy Coverage – Summary

Bloom's Level	Level Name	CO(s)	Sample Activities
L3	Apply	**CO1, CO2**	Molecular orbital energy diagram construction; semiconductor band gap and doping calculations.
L4	Analyze	**CO3, CO4**	Battery electrochemical reaction analysis; corrosion mechanism case studies for PCB failure modes.
L5/L6	Evaluate/Create	**CO5**	Nanomaterial design for a given computing application; AI-assisted property prediction project.

LEARNING RESOURCES

Textbooks

1. Jain, P. C., & Jain, M. (2020). Engineering Chemistry (17th ed.). Dhanpat Rai Publishing Company.
2. Shika Agarwal. (2026). Engineering Chemistry (5th ed.). Cambridge University Press.
3. K. Sesha Maheswaramma, Mridula Chugh, “Engineering Chemistry”, Pearson

Publications Pvt., Ltd., 2016 (ISBN 978-93-325-7118-1) & (ISBN 978-93-895-8839-2) 2019.

Reference Books

1. Linden, D., & Reddy, T. B. (Eds.). (2001). Handbook of Batteries (3rd ed.). McGraw-Hill.
2. Rao, C. N. R., Müller, A., & Cheetham, A. K. (Eds.). (2007). The Chemistry of Nanomaterials: Synthesis, Properties and Applications (Vols. 1–2). Wiley-VCH.
3. Poole, C. P., & Owens, F. J. (2003). Introduction to Nanotechnology. John Wiley & Sons.
4. Sharma, B. K., & Gaur, H. (2019). Engineering Chemistry. Krishna Prakashan Media.
5. Dara, S. S., & Umare, S. S. (2018). Engineering Chemistry (15th ed.). S. Chand Publishing.

Online Resources

- NPTEL Engineering Chemistry: https://nptel.ac.in/
- IBM Qiskit Learning: https://learning.quantum.ibm.com/
- Google Colab: https://colab.research.google.com/
- OpenCV Python Documentation: https://docs.opencv.org/
- Materials Project (open nanomaterials database): https://materialsproject.org/

NPTEL / SWAYAM COURSES

1. Concept of Chemistry - IIT Madras
2. Engineering Chemistry I – IIT Madras
3. Engineering Chemistry I – IIT Bombay
4. Elementary Electrochemistry – IISER Mohali

I Year - I Semester

Course Code	LINEAR ALGEBRA AND CALCULUS	L	T	Pr	P	Credits
26BST02	**(Common to all branches)**	3	0	2	0	4
Prerequisite:	Higher Secondary Mathematics					

Course Objectives

- Build strong foundations in linear algebra and multivariable calculus with direct engineering relevance.
- Integrate analytical methods with AI-assisted computational tools for visualization, simulation, and verification.
- Develop outcome-oriented problem-solving skills aligned with NEP 2020 and modern engineering applications.
- Apply matrix methods, Eigen analysis, and multivariable calculus to solve engineering problems.
- Use computational tools for approximation, optimization, differentiation, and integration.

Course Outcomes (COs)

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

CO	Course Outcome Statement	Bloom's Level
CO1	Apply matrix algebra techniques to solve systems of linear equations and analyze matrix properties relevant to engineering models.	**L3 (Apply)**
CO2	Analyze Eigen value problems, Diagonalization, and quadratic forms to interpret linear transformations and engineering systems.	**L4 (Analyze)**
CO3	Apply mean value theorems, Taylor and Maclaurin series, and curvature concepts to approximate and interpret engineering functions.	**L3 (Apply)**
CO4	Analyze multivariable functions using partial derivatives, Jacobians, gradients, and optimization techniques.	**L4 (Analyze)**
CO5	Evaluate areas, volumes, and coordinate transformations using multiple integrals in engineering contexts.	**L5 (Evaluate)**

Course Content

UNIT – 1	Matrix Algebra and Linear Systems	9 Hrs
Content	• Matrices, types of matrices, operations, transpose inverse, symmetric and skew-symmetric matrices; determinant properties; rank of a matrix; echelon and normal forms. • System of linear equations; consistency conditions; Gauss elimination and Gauss-Jordan methods; iterative methods: Jacobi and Gauss-Seidel. • Cramer's rule; matrix inversion method; applications to engineering systems.	

AI Integration	Python-based solution of large linear systems using NumPy; visualization of convergence in iterative methods; AI-assisted debugging of matrix algorithms; using Chat GPT/Copilot to explain row operations and rank reduction. Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib) on Google Colab; SymPy; Jupyter Notebook; Scilab.

UNIT – 2	**Eigen values, Eigenvectors, Diagonalization and Quadratic Forms**	**9 Hrs**
Content	• Eigen values and eigenvectors; characteristic equation; Cayley-Hamilton theorem; Diagonalization; similarity transformation; powers and inverse of matrices using Cayley-Hamilton theorem. • Quadratic forms; canonical form; reduction by orthogonal transformation; rank, index, signature; positive definite forms; orthogonal, Hermitian, skew-Hermitian and unitary matrices.	
AI Integration	AI-assisted Eigen value computation and verification; Python/SciPy visualization of eigen modes and quadratic surfaces; use of AI tools to explain Diagonalization and matrix classification; numerical checking of Cayley-Hamilton theorem. Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib); SymPy; GNU Octave; Google Colab.	
UNIT – 3	**Single Variable Calculus and Series Expansions**	**9 Hrs**
Content	• Rolle's theorem; Lagrange's mean value theorem; Cauchy's mean value theorem; Taylor's and Maclaurin's series with remainder; approximation of functions; error estimation. • Curvature of plane curves; radius and centre of curvature; circle of curvature; applications to engineering approximations and curve design.	
AI Integration	SymPy-based Taylor and Maclaurin expansion generation; plotting truncation error and convergence; AI-supported derivation checking for mean value theorems; curvature plot visualization and interpretation. Free/Open-Source AI Tools: Python (SymPy, NumPy, Matplotlib); Google Colab; Jupyter Notebook; WolframAlpha free tier.	
UNIT – 4	**Multivariable Differentiation and Optimization**	**9 Hrs**
Content	• Functions of several variables; limits and continuity; partial derivatives; higher order partial derivatives; total derivative; chain rule; gradient; tangent plane and normal line. • Jacobians; functional dependence; Taylor expansion of two-variable functions; maxima and minima; second derivative test; Lagrange multipliers and constrained optimization;	
AI Integration	Gradient field visualization using Python; AI-assisted Jacobian computation and verification; constrained optimization using SciPy; use of AI tools to interpret saddle points, extrema, and optimization constraints. Free/Open-Source AI Tools: Python (NumPy, SciPy, SymPy, Matplotlib); Google Colab; Jupyter Notebook; SciPy optimize module.	

UNIT – 5	Multiple Integrals and Coordinate Transformations	9 Hrs
Content	• Double integrals; evaluation over type I and type II regions; change of order of integration; triple integrals; change of variables; polar, cylindrical and spherical coordinates. • Area and volume computation; mass, centroid and moment calculations; engineering applications of multiple integration.	
AI Integration	Numerical evaluation of double and triple integrals using SciPy; 2D and 3D region visualization; AI-assisted generation of integration code; interpreting Jacobian as geometric scaling factor in transformed coordinates. Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib); Google Colab; SymPy; Jupyter Notebook; 3D plotting tools.	

Bloom's Taxonomy Coverage – Summary

Bloom's Level	Level Name	CO(s)	Sample Activities
L3	Apply	**CO1, CO3**	Gauss elimination and Gauss-Jordan problem-solving; Taylor series approximation exercises.
L4	Analyze	**CO2, CO4**	Eigen value and Diagonalization analysis; Jacobian and constrained optimization problems.
L5	Evaluate	**CO5**	Multiple integral evaluation for area/volume/Centroid computation; coordinate transformation assessment.

CO – PO Articulation Matrix (NBA)

Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	2	2	1	-	-	-	1	2	-
CO2	3	3	3	2	1	-	-	-	1	2	-
CO3	3	2	2	2	1	-	-	-	-	2	-
CO4	3	3	3	3	2	-	-	-	1	2	-
CO5	3	3	2	3	3	-	1	1	2	2	1

CO-PO Correlation Highlights	CO1 and CO2 strongly align with PO1 (Engineering Knowledge) and PO2 (Problem Analysis) through matrix and eigenvalue problem-solving. CO4 and CO5 map to PO3 and PO4 through optimization and multiple integral applications. PO5 (Modern Tools) is addressed throughout via Python, SymPy, and Google Colab integration. CO5 additionally links to PO7 and PO8 through engineering field interpretation, and strongly to PO12 (Lifelong Learning) through continued computational skill development beyond the course.

LEARNING RESOURCES

Textbooks

1.Grewal, B. S. (2017). Higher Engineering Mathematics (44th ed.). Khanna Publishers.
2.Kreyszig, E. (2018). Advanced Engineering Mathematics (10th ed.). Wiley.

Reference Books

1.Thomas, G. B., Weir, M. D., & Hass, J. (2018). Thomas' Calculus (14th ed.). Pearson.
2.Jain, R. K., & Iyengar, S. R. K. (2021). Advanced Engineering Mathematics (5th ed.). Alpha Science International.
3.Strang, G. (2016). Introduction to Linear Algebra (5th ed.). Wellesley-Cambridge Press.
4.Lay, D. C., Lay, S. R., & McDonald, J. J. (2015). Linear Algebra and Its Applications (5th ed.). Pearson.
5.Arfken, G. B., Weber, H. J., & Harris, F. E. (2012). Mathematical Methods for Physicists (7th ed.). Academic Press.
6.Zill, D. G., & Wright, W. S. (2018). Advanced Engineering Mathematics (6th ed.). Jones & Bartlett.

Online Resources

- NPTEL: Linear Algebra – https://nptel.ac.in
- MIT OpenCourseWare – Linear Algebra: https://ocw.mit.edu
- MIT OpenCourseWare – Multivariable Calculus: https://ocw.mit.edu
- Python SymPy Documentation: https://docs.sympy.org
  - Google Colab for Python: https://colab.research.google.com

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, Matplotlib) – FREE on Google Colab for matrix computation, visualization, and numerical analysis.
- SymPy (free, open-source) – For symbolic algebra, calculus, matrix operations, and verification.
- scikit-learn (free, open-source) – For regression, classification, and exploratory predictive modeling.
- Jupyter Notebook (free, open-source) – For interactive computation and lab documentation.

- Open CV (free, open-source) – For image-based experimentation and visualization support where needed.
- GNU Octave (free, open-source) – MATLAB-like numerical computing for matrix and calculus workflows.
- Tensor Flow / Keras (free, open-source) – For optional AI demonstrations in regression and classification tasks.

I B.Tech I Semester

Course Code	DESIGN THINKING	L	T	Pr	P	Credits
26MET02	**Common to All Branches**	1	0	4	0	3

Course Overview

Design Thinking is a human-centred innovation methodology that equips students with a structured approach to understanding users, reframing complex problems, and developing creative solutions. This course integrates Artificial Intelligence tools at each stage of the design cycle—empathise, define, ideate, prototype, and test—enabling students to harness AI not merely as a tool but as a creative collaborator. Students work on real-world projects and submit iterative prototypes.

Detailed Syllabus

UNIT 1: Introduction to Design Thinking

Core Concepts:

- Design Thinking: Definition, origin (Stanford d.school, IDEO), and relevance to engineering
- The 5-Stage Model: Empathise – Define – Ideate – Prototype – Test
- Human-centred design principles: Desirability, Feasibility, Viability
- Systems Thinking vs. Linear Thinking: Understanding complexity
- Case Studies: Successful design thinking applications (IDEO shopping cart, Stanford Mobile Clinic)

AI Integration:

- AI Tool: ChatGPT / Copilot – use AI to explain design thinking in different domains (healthcare, education, agriculture)
- Activity: Use MindMeister (free, open-source alternative: FreeMind) to create a mindmap of the 5-stage process
- AI Video Analysis: Watch an IDEO documentary and use AI to summarise key DT principles observed

UNIT 2: Empathise and Define Stages

Core Concepts:

- Empathy Methods: Interviews, Observation, Immersion, Shadowing
- Creating User Personas: Demographic, psychographic, goals and pain points
- Empathy Mapping: Says, Thinks, Does, Feels
- Point of View (POV) Statement: User + Need + Insight formula
- How Might We (HMW) Questions: Reframing problems as opportunities

AI Integration:

- AI Tool: Miro (free plan) – create digital empathy maps and POV statements collaboratively
- AI Tool: ChatGPT – generate diverse user persona profiles for a given problem context
- Activity: Interview 5 peers about a campus problem; use AI to analyse interview transcripts and extract themes
- AI Tool: Dovetail (free plan) – affinity mapping of qualitative research data

UNIT 3: Ideation Stage

Core Concepts:

- Creative Thinking Techniques: Brainstorming, Brainwriting, SCAMPER, Random Word Association
- Lateral Thinking: Edward de Bono's Six Thinking Hats
- Affinity Diagramming: Grouping ideas by themes
- Prioritisation Matrix: Impact vs. Effort, Dot Voting
- Concept Selection: Morphological charts and Pugh Matrix

AI Integration:

- AI Tool: ChatGPT – use 'expand my idea' prompts to generate 20 variations of a concept
- AI Tool: IdeaFlip (free) / Miro – digital brainstorming and affinity diagramming
- AI Tool: Midjourney (free trial) / DALL-E (free credits) – generate visual concept sketches from text descriptions
- Activity: Run a 30-minute AI-augmented brainstorming session and compare idea quality with pure human brainstorming

UNIT 4: Prototyping

Core Concepts:

- Principles of Prototyping: Low-fidelity vs. High-fidelity prototypes
- Paper Prototyping: Sketching UI and physical models
- Digital Wireframing: Introduction to wireframes and mockups
- 3D Prototyping Concepts: Introduction to 3D printing and rapid prototyping
- Role Plays and Experience Prototyping: Service design scenarios

AI Integration:

- AI Tool: Figma (free plan) – create digital wireframes and interactive mockups
- AI Tool: Uizard (free plan, AI-powered) – convert hand-drawn sketches to digital wireframes using AI
- AI Tool: Tinkercad (free, web-based) – create simple 3D models for physical prototyping
- Activity: Build a paper prototype for a chosen problem and photograph for digital documentation

UNIT 5: Testing, Iteration and AI in Design

Core Concepts:

- User Testing Methods: Think-aloud protocol, A/B testing, Usability testing
- Feedback Analysis: Affinity mapping of test feedback
- Iteration Cycle: Incorporating feedback and pivoting
- Design Presentation: Pitching to stakeholders (elevator pitch, demo day format)
- AI in Design: Overview of AI-assisted design (generative design, co-creation with AI)

AI Integration:

- AI Tool: Maze (free plan) – conduct remote usability tests on Figma prototypes
- AI Tool: Chat GPT – analyse collected user feedback and generate insights and improvement suggestions

- Activity: Present a final prototype to a panel; use AI to prepare pitch slides in Canva
- Reflection: Create a Design Thinking journal documenting each stage with AI tool usage notes

Course Outcomes

CO	Course Outcome Statement	Bloom's Level	Bloom's Domain
CO1	Recall and understand the philosophy, principles, and stages of the Design Thinking process and its relevance to human-centred problem solving.	L1 – Remember / L2 – Understand	Cognitive
CO2	Apply empathy tools (interviews, observation, personas) and problem-framing techniques to real-world engineering scenarios.	L3 – Apply	Cognitive & Affective
CO3	Analyse user needs and problem statements using the 'How Might We' framework and AI-powered insight generation tools.	L4 – Analyse	Cognitive
CO4	Evaluate and select the most viable prototype solution through structured testing, user feedback, and AI-assisted evaluation rubrics.	L5 – Evaluate	Cognitive
CO5	Create innovative prototypes and pitch solutions to identified problems using AI ideation tools and digital prototyping platforms.	L6 – Create	Cognitive & Psychomotor

Text Books

1. T1. Brown, Tim. (2019). Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation. HarperBusiness, New York.
2. T2. Lewrick, Michael, Link, Patrick & Leifer, Larry. (2018). The Design Thinking Playbook. Wiley, New Jersey.

Reference Books

1. R1. IDEO.org. (2015). The Field Guide to Human-Centered Design. IDEO.org, San Francisco.
2. R2. Kelley, Tom & Kelley, David. (2013). Creative Confidence: Unleashing the Creative Potential Within Us All. Crown Business.
3. R3. Liedtka, Jeanne & Ogilvie, Tim. (2011). Designing for Growth: A Design Thinking Tool Kit for Managers. Columbia Business School Press.
4. R4. Plattner, Hasso, Meinel, Christoph & Leifer, Larry (Eds.). (2018). Design Thinking Research. Springer.

5. R5. Kumar, Vijay. (2012). 101 Design Methods: A Structured Approach for Driving Innovation. Wiley.
6. R6. Stickdorn, Marc & Schneider, Jakob. (2021). This Is Service Design Thinking (Expanded ed.). BIS Publishers.

Web Resources

- Stanford d.school Resources: dschool.stanford.edu/resources
- IDEO Design Kit: www.designkit.org/methods
- MIT OpenCourseWare – Design Thinking: ocw.mit.edu
- Interaction Design Foundation: www.interaction-design.org
- NPTEL Design Thinking: swayam.gov.in

YouTube / Video Resources

- YouTube: Stanford d.school – Design Thinking Bootcamp – www.youtube.com/@stanforddschool
- YouTube: IDEO U – www.youtube.com/@ideou
- YouTube: Crash Course – Design Thinking Series – www.youtube.com/@crashcourse
- YouTube: AJ&Smart – Design Sprint tutorials – www.youtube.com/@AJSmart
- YouTube: NPTEL Design Thinking – www.youtube.com/c/nptel

I B.Tech I Semester

Course Code	COMPUTATIONAL THINKING AND PROBLEM SOLVING WITH PYTHON	L	T	Pr	P	Credits
26CST01	(Common to all branches)	3	0	0	0	3

Course Objectives:

- Introduce students to computational thinking and problem-solving techniques.
- Develop the ability to design algorithms, flowcharts, and pseudocode.
- Provide foundational knowledge of Python programming — data types, operators, and I/O.
- Enable students to implement logic using control structures and data structures.
- Develop skills in modular programming, file handling, and basic OOP concepts.

Course Outcomes:

After completion of the course, students will be able to

CO	Course Outcome Statement	Bloom's Taxonomy Level
CO1	Apply computational thinking concepts to design algorithms, flowcharts, and basic Python programs. (L3)	L3 (Apply)
CO2	Develop programs using decision-making and looping constructs to solve logical problems. (L3)	L3 (Apply)
CO3	Analyse and manipulate data using strings, lists, tuples, and sets for problem solving. (L4)	L4 (Analyze)
CO4	Design modular programs using functions, recursion, and dictionaries to improve code reusability. (L4)	L4 (Analyze)
CO5	Implement file handling, exception handling, and basic object-oriented programming concepts in Python applications. (L3)	L3 (Apply)

UNIT – I	COMPUTATIONAL THINKING AND PROGRAMMING BASICS	9 Hrs

Content	• Computational thinking: characteristics, problem-solving strategies, steps in problem solving; algorithms — definition and properties; flowcharts — symbols and construction; pseudocode — writing and conversion; abstraction, decomposition, pattern recognition, algorithm efficiency basics. • Introduction to Python: installation and execution environment; variables, identifiers, keywords; data types, type conversion; input and output statements; expressions and operators — arithmetic, relational, logical, assignment; operator precedence. **Source: TB1: Chapters 1–3 (Spronck — computational thinking, algorithms, flowcharts, Python basics); TB2: Chapters 1–2 (Thareja — introduction to programming, Python fundamentals)**
AI Integration	Use Flowgorithm to design and simulate algorithms (e.g., largest of 3 numbers); draw flowcharts using Lucidchart; use Algorithm Visualizer to trace algorithm execution; use ChatGPT to convert pseudocode to Python and verify output.

UNIT – II	DECISION MAKING AND LOOPING	9 Hrs

Content	• Decision control statements: Boolean expressions; if, if-else, if-elif-else, nested if; conditional expressions (ternary operator). • Looping statements: while loop, for loop, iteration techniques, nested loops, infinite loops; loop control — break, continue, pass; else with loops. • Practical problem solving: prime number check, pattern programs using nested loops, menu-driven programs using control statements. **Source: TB2: Chapters 4–5 (Thareja — decision making, loops, control statements); TB1: Chapters 4–5 (Spronck — conditions, loops, iteration)**
AI Integration	Use Python Tutor to visualize if-else and loop execution step by step; use Thonny to debug loop-based programs; use Replit for online coding practice; use ChatGPT to generate test cases for control flow programs.

UNIT – III	**STRINGS AND DATA STRUCTURES**	**9 Hrs**

Content	• Strings: representation, indexing, slicing, operations, built-in functions and methods. • Lists: creation, indexing and slicing, operations, functions, methods, nested lists. • Tuples: creation, operations, packing and unpacking. • Sets: creation, set operations — union, intersection, difference; frozen sets. **Source: TB2: Chapters 6–8 (Thareja — strings, lists, tuples, sets, dictionaries); TB1: Chapters 6–7 (Spronck — strings, lists, tuples)**
AI Integration	Use NLTK and TextBlob to perform text analysis (sentiment, word frequency) on strings; use Pandas and NumPy for list and array operations; use ChatGPT to explain string slicing and list comprehensions with concrete examples.

UNIT – IV	**FUNCTIONS AND PROBLEM SOLVING**	**9 Hrs**

Content	• Dictionaries: creation, operations, methods; dictionary-based applications. • Functions: built-in and user-defined functions; function definition and calling; arguments — positional, keyword, default, variable-length; scope of variables (local and global). • Recursion: recursive functions — factorial, Fibonacci; lambda functions (anonymous functions); applications of functions in problem solving. **Source: TB2: Chapters 9–10 (Thareja — functions, recursion, lambda, dictionaries); TB1: Chapters 8–9 (Spronck — functions, recursion, problem solving)**
AI Integration	Use Jupyter Notebook for interactive function experiments; use SymPy for symbolic mathematical expressions; use PyCharm for modular program development; use ChatGPT to trace recursive function calls step by step.

UNIT – V	**FILE HANDLING, EXCEPTION HANDLING, AND OOP**	**9 Hrs**

Content	• Modules and packages: creating and importing modules; standard library modules. • File handling: opening, reading, writing, and closing files; file modes; working with text files; processing CSV and Excel files.

	• Exception handling: types of errors (syntax, runtime, logical); try, except, finally blocks; raising exceptions. • Introduction to OOP: classes, objects, attributes, methods, constructors, self keyword; basic applications of OOP in Python. **Source: TB2: Chapters 11–13 (Thareja — modules, file handling, exception handling, OOP); TB1: Chapter 10 (Spronck — file handling, exceptions, object-oriented programming)**
AI Integration	Use Pandas and OpenPyXL to read and write CSV/Excel files; use Pytest for testing exception handling; use ChatGPT to explain OOP class design with real-world examples; use Django for a simple mini web application as a capstone project.

Bloom's Taxonomy Coverage – Summary

Bloom's Level	Level Name	CO(s)	Sample Activities / Assessments
L1–L2	Remember / Understand	**CO1**	Define algorithm, pseudocode, flowchart; describe data types and operators.
L3	Apply	**CO1–CO5**	Write Python programs using control structures, functions, and file handling.
L4	Analyse	**CO3, CO4**	Compare list vs. tuple vs. set; analyse recursive vs. iterative solutions.
L5	Evaluate	**CO5**	Evaluate exception handling strategies for a given fault scenario.
L6	Create	**CO4, CO5**	Design a modular OOP-based Python application for a real-world problem.

CO – PO Articulation Matrix (NBA)

Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	–	–	2	–	–	–	1	–	1
CO2	3	3	2	–	2	–	–	–	–	–	1
CO3	3	3	2	2	3	–	–	–	–	–	1
CO4	3	3	3	2	3	–	–	–	–	1	1
CO5	3	2	2	2	3	1	2	–	–	1	1

LEARNING RESOURCES

Textbooks

1. TB1: Pieter Spronck, Computational Thinking with Python, (course textbook — covers computational thinking, algorithms, and Python programming).
2. TB2: Reema Thareja, Programming in Python, Oxford University Press. (Primary implementation reference — all five units.)

Reference Books

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, O'Reilly Media. (Free at greenteapress.com)
2. Eric Matthes, Python Crash Course, 2nd Edition, No Starch Press.

Online Resources

- Algorithm & Flowchart: Flowgorithm – flowgorithm.org | Lucidchart – lucidchart.com | Algorithm Visualizer – algorithm-visualizer.org
- Programming & Debugging: Python Tutor – pythontutor.com | Thonny – thonny.org | Replit – replit.com
- Data & Scientific: Pandas – pandas.pydata.org | NumPy – numpy.org | SymPy – sympy.org
- NLP & Text: NLTK – nltk.org | TextBlob – textblob.readthedocs.io
- Development: Jupyter Notebook – jupyter.org | PyCharm – jetbrains.com/pycharm | OpenPyXL – openpyxl.readthedocs.io
- NPTEL – Problem Solving through Programming in C / Python – nptel.ac.in

I B.Tech I Semester

Course Code	COMMUNICATIVE ENGLISH (Common to All Branches)	L	T	Pr	P	Credits
26BST01	(Common to all branches)	2	0	0	0	2

Course Objectives:

- Develop strong foundations in English grammar, vocabulary, phonetics, and communication models essential for professional contexts.
- Develop listening, speaking, reading, and writing skills using structured activities and AI-powered tools.
- Enable students to compose professional emails, reports, and presentations using AI writing assistants.
- Develop critical analysis of communication styles, barriers, and discourse structures using AI tools.
- Cultivate responsible AI usage for language learning — verifying AI outputs, maintaining academic integrity, and using AI as an aid not a substitute.

Course Outcomes:

After completion of the course, students will be able to

CO	Course Outcome Statement	Bloom's Taxonomy Level
CO1	Identify grammatical structures, phonetic patterns, and professional vocabulary, and recall communication frameworks using AI-assisted tools.	L1 (Remember) L2 (Understand)
CO2	Comprehend and interpret written and spoken communication models, and summarize content using AI paraphrasing and summarization tools.	L2 (Understand)
CO3	Compose professional emails, reports, and presentations using AI writing assistants such as ChatGPT, Grammarly, and Hemingway Editor.	L3 (Apply)
CO4	Analyse discourse structures, identify communication barriers, and use AI-driven sentiment analysis tools to evaluate message effectiveness.	L4 (Analyze)
CO5	Evaluate communication samples using AI grammar checkers, readability scorers, and peer-evaluation rubrics aligned to professional standards.	L5 (Evaluate)
CO6	Design and deliver professional communication artifacts — presentations, e-portfolios, and research reports — integrating AI tools for structure, style, and refinement.	L6 (Create)

UNIT – I	FOUNDATIONS OF ENGLISH COMMUNICATION	9 Hrs
Content	Basics of Communication: process, types (verbal/non-verbal), and barriers. Parts of Speech: nouns, pronouns, verbs, adjectives, adverbs. Tenses: simple, continuous, perfect, and perfect continuous. Subject-verb agreement and correction of sentences.	

AI Integration	Grammar explanation, error identification, and instant Q&A on language rules, Real-time grammar, spelling, and punctuation correction with detailed explanations, Phonetics practice – voice recording for pronunciation and intonation feedback, Word-of-the-day, etymology, and interactive vocabulary quizzes. Free/Open-Source AI Tools: Grammarly (free); ChatGPT / Gemini (free tiers); Google Speech-to-Text (free); Merriam-Webster Vocabulary Builder (free)

UNIT – II	**LISTENING AND SPEAKING SKILLS**	**9 Hrs**

Content	Principles of Effective Speaking and Active Listening. Group Discussion (GD): techniques, etiquette, and evaluation criteria. Presentation Skills: design principles, slide preparation, and delivery techniques. Debate: argument construction, evidence use, and rebuttal techniques.
AI Integration	AI-powered presentation design – structured, visual, professional slides, Fluency analysis, filler-word detection, pacing, and confidence scoring, Auto-transcription and keyword analysis of spoken communication, Visual communication design – infographics, posters, and slide aesthetics Free/Open-Source AI Tools: Gamma.app (free); Orai AI Speech Coach (free); Otter.ai (free tier); Canva AI (free)

UNIT – III	**READING COMPREHENSION & WRITING SKILLS**	**9 Hrs**

Content	Reading Strategies: skimming, scanning, intensive and extensive reading. Comprehension passages, synonyms, antonyms, words often confused, homonyms, homophones, and homographs. Paragraph Writing: topic sentence, supporting details, clincher sentence. Creative Writing — features and techniques — movie reviews and blogs.
AI Integration	AI paraphrasing tool – rewrite passages at varying complexity levels, Readability scoring, passive voice detection, and sentence simplification, Essay outline generation, argument structuring, and content brainstorming, Vocabulary simplification for improving comprehension of complex texts Free/Open-Source AI Tools: QuillBot (free); Hemingway Editor (free online); ChatGPT (free); Rewordify (free)

UNIT – IV	**PROFESSIONAL & BUSINESS COMMUNICATION**	**9 Hrs**

Content	Email Writing and Letter Writing: formal and informal. Short Report Writing: structure, types (formal/informal), and format. Note-Making and Summarization Techniques. Resume / CV Writing and LinkedIn Profile Optimization.

AI Integration	Draft and refine professional emails, cover letters, reports, and proposals, AI-powered resume builder with ATS-optimized templates, Report structuring, agenda writing, and knowledge management, Tone detection, formality check, and professional register evaluation Free/Open-Source AI Tools: ChatGPT (free); Kickresume (free tier); Grammarly (free); Notion AI (free tier)

UNIT – V	USE OF AI IN LANGUAGE LEARNING	9 Hrs

Content	Artificial Intelligence in Communication: meaning, basics, benefits and limitations of AI in human interaction. AI chatbots and virtual assistants in education: grammar correction, vocabulary building, AI-based speaking and pronunciation tools, writing assistants, e-portfolio creation. AI in Global Communication: automated translation, summarizing large texts, finding information in different languages. Digital Communication and AI Ethics: responsible use of AI in education, plagiarism and academic honesty, fake information and AI-generated content, privacy, bias, and ethical communication online.
AI Integration	Plagiarism detection, paraphrase checker, and academic integrity analysis, High-accuracy translation supporting intercultural communication understanding, AI literacy modules – understanding AI bias, ethics, and responsible usage, E-portfolio creation platform for showcasing communication artifacts Free/Open-Source AI Tools: Copyleaks / Turnitin (institutional access); DeepL Translator (free); Wix (free); Adobe Portfolio (free for students); ChatGPT / Gemini (free tiers).

LEARNING RESOURCES

Textbooks

1. Raman M. and Sharma S., Technical Communication: Principles and Practice, Oxford University Press, 4th Edition, 2022.
2. Shetty J. and Pareek R., Communicative English for Engineers and Professionals, Wiley India, 3rd Edition, 2021.

Reference Books

1. Rizvi M.A., Effective Technical Communication, Tata McGraw-Hill, 2nd Edition, 2018.
2. Murphy H.A., Effective Business Communication, McGraw-Hill Education, 7th Edition, 2019.
3.Wren P.C. and Martin H., High School English Grammar and Composition, S. Chand, Revised Edition, 2019.

Online Resources

- British Council – Learn English – https://learnenglish.britishcouncil.org
- NPTEL – Communication Skills – https://nptel.ac.in/courses/109104101
- Purdue OWL (Online Writing Lab) – https://owl.purdue.edu
- BBC Learning English – https://www.bbc.co.uk/learningenglish

I B.Tech I Semester

Course Code	APPLIED CHEMISTRY LAB FOR COMPUTER SCIENCE	L	T	Pr	P	Credits
26BSP06		0	0	0	3	1.5
Prerequisite:	Applied Chemistry for Computer Science					

Lab Course Outcomes

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

CO	Course Outcome Statement	Bloom's Level
CO1	Perform redox and conductometric titrations to estimate concentrations of analytes in solution.	**L3**
CO2	Determine corrosion rates and analyze environmental factors affecting corrosion of metals.	**L3–L4**
CO3	Apply potentiometric and spectrophotometric techniques for quantitative chemical analysis.	**L3**
CO4	Identify functional groups in organic compounds using IR spectroscopy.	**L3**
CO5	Synthesize a polymer and use AI/Python-based data analysis for predictive modelling of material behaviour.	**L3–L4**

List of Experiments

1. Estimation of ferrous iron (Fe^{2+}) using potassium permanganate ($KMnO_4$) -redox titration.
2. Determination of strength of acid (H_2SO_4) in battery using sodium hydroxide (NaOH).
3. Determination of rate of corrosion (mild steel or copper) - calculate the corrosion rate using the weight-loss method. Document observations on surface changes.
4. Study of environmental factors (humidity, salinity, acidity) which influence the rate of corrosion.
5. Conductometric titration of strong acid vs strong base (HCl vs NaOH).
6. Conductometric titration of weak acid vs strong base (CH_3COOH vs NaOH).
7. Potentiometry (redox titration) — estimation of iron (Fe^{2+}) using potassium dichromate ($K_2Cr_2O_7$).
8. Spectrophotometry — Determination of (λ_{max}) of $KMnO_4$ and verification of Beer-Lambert's law.

9. Identification of functional groups in organic compound by infra-red (IR) spectroscopy.
10. Preparation of polymer (phenol-formaldehyde — Bakelite or any other polymer).

AI Integration (Lab): Data analysis using Python (NumPy, Pandas, Matplotlib) for titration curve plotting and equivalence point detection. Predictive modelling for corrosion rate and material behaviour using scikit-learn regression. TensorFlow for optional IR spectrum pattern classification. OpenCV for colorimetric image-based concentration estimation. Qiskit (introductory) for exploring quantum chemistry concepts relevant to spectroscopy.

LEARNING RESOURCES

Textbooks

1. Jain, P. C., & Jain, M. (2020). Engineering Chemistry (17th ed.). Dhanpat Rai Publishing.
2. Shika Agarwal (2026). Engineering Chemistry (5th ed.). Cambridge University Press.

Reference Books

1. Dara, S. S., & Umare, S. S. (2018). Engineering Chemistry (15th ed.). S. Chand Publishing.

Online Resources

- IEEE Xplore Digital Library
- Semiconductor Industry Association resources
- NPTEL

NPTEL / SWAYAM COURSES

1. Analytical Chemistry – IIT Kharagpur
2. Quantitative methods in Chemistry – IISER Bhopal

Assessment Pattern (R26)

Assessment Component	Frequency / Duration	Marks	COs Assessed
Lab Internal (Record + Viva)	Continuous	25	CO1–CO4
Mid-Term Practical Test	1 test	15	CO1, CO2, CO3
AI Lab Exercises	3 exercises	10	CO5
End Semester Practical Exam	3 hrs	35	CO1–CO4
End Semester Viva Voce	30 min	15	All COs
TOTAL		**100**	

Note: Lab record to be submitted for every experiment at the next session. AI exercises to be submitted as Google Colab .ipynb notebooks.

I B.Tech I Semester

26CSP01	Computational Thinking and Problem Solving with Python Lab	L	T	Pr	P	C
		0	0	0	3	1.5

Course Objectives

- Introduce students to computational thinking and problem-solving techniques.
- Develop the ability to design algorithms, flowcharts, and pseudocode.
- Provide foundational knowledge of Python programming including data types, operators, and input/output.
- Enable students to implement logic using control structures and Python data structures.
- Develop skills in modular programming, file handling, and basic object-oriented programming.

List of Experiments

MODULE I: COMPUTATIONAL THINKING & PROGRAMMING BASICS

Topics: Computational thinking — decomposition and abstraction; algorithms, flowcharts, and pseudocode; Python environment setup; variables, data types, input/output, and operators; basic computational problems.

AI Integration: Generate alternate solutions and test inputs; verify flowchart correctness; compare solution efficiency; explain type conversion.

Suggested Tools: Python, Flowgorithm, draw.io AI, ChatGPT, Microsoft Copilot

Experiments:

- **Experiment 1: Problem Solving using Computational Thinking —** Select three daily-life problems; apply decomposition and abstraction; write algorithms, draw flowcharts, and convert them into pseudocode.
- **Experiment 2: Flowchart Design and Algorithm Simulation —** Design flowcharts for largest of three numbers, student grade calculation, and electricity-bill generation; simulate execution and verify outputs.
- **Experiment 3: Python Programming Fundamentals —** Install the Python environment; write programs using variables, data types, input/output, and operators; solve simple interest, area, and unit-conversion problems.

MODULE II: DECISION MAKING & LOOPING

Topics: Decision-control statements; looping and iterative problem solving; nested loops and pattern generation.

AI Integration: Generate boundary and test cases; verify decision logic; trace loop execution; compare loop alternatives; optimize nested loops.

Suggested Tools: Python, Python Tutor, ChatGPT, Microsoft Copilot

Experiments:

- **Experiment 4: Decision Control Statements —** Develop Python programs for largest among three numbers, grade calculation, leap-year checking, and a menu-driven calculator.

- **Experiment 5: Looping and Iterative Problem Solving —** Implement programs for prime-number generation, Armstrong numbers, factorial computation, and multiplication tables.
- **Experiment 6: Nested Loops and Pattern Generation —** Generate number, pyramid, diamond, and character patterns.

MODULE III: STRINGS & DATA STRUCTURES

Topics: String processing; lists and nested lists; tuples and sets.

AI Integration: Generate string test cases and sample datasets; explain slicing and immutable structures; compare list approaches and suggest optimizations.

Suggested Tools: Python, Python Tutor, Algorithm Visualizer, ChatGPT

Experiments:

- **Experiment 7: String Processing Applications —** Develop programs for string reversal, palindrome checking, frequency counting, and word-occurrence analysis.
- **Experiment 8: List and Nested List Operations —** Perform sorting, searching, list merging, and matrix operations using nested lists.
- **Experiment 9: Tuple and Set Applications —** Implement tuple packing/unpacking, student-record storage, and set union, intersection, and difference operations.

MODULE IV: FUNCTIONS & PROBLEM SOLVING

Topics: User-defined functions; recursion; dictionaries and lambda functions.

AI Integration: Generate function examples and datasets; verify outputs; generate recursion trees; compare iterative and recursive solutions; explain lambda execution.

Suggested Tools: Python, Python Tutor, Algorithm Visualizer, ChatGPT

Experiments:

- **Experiment 10: User-Defined Functions —** Implement positional, keyword, default, and variable-length arguments.
- **Experiment 11: Recursive Problem Solving —** Develop recursive solutions for factorial, Fibonacci, GCD, and Tower of Hanoi.
- **Experiment 12: Dictionary and Lambda Applications —** Implement a student database, dictionary operations, dictionary sorting, and lambda-based computations.

MODULE V: FILE HANDLING, EXCEPTION HANDLING & OOP

Topics: File processing and data storage; exception handling and testing; object-oriented programming — classes, objects, constructors, methods.

AI Integration: Generate datasets and error scenarios; verify file outputs; recommend exception strategies; generate class diagrams and suggest object interactions.

Suggested Tools: Python, ChatGPT, Microsoft Copilot

Experiments:

- **Experiment 13: File Processing and Data Storage —** Develop programs to create and read text files, count words, and process CSV data.
- **Experiment 14: Exception Handling and Testing —** Implement try-except, finally, and custom exceptions; apply exception handling to division operations, file access, and input validation.
- **Experiment 15: Object-Oriented Programming Mini Project —** Develop a Python application using classes, objects, constructors, and methods.

Suggested Mini Projects:

1. Student Management System
2. Library Management System
3. Banking System

Course Outcomes (Cos)

CO	Course Outcome Statement	Bloom's Level
CO1	Apply computational thinking concepts to design algorithms, flowcharts, and Python programs.	L3 (Apply)
CO2	Develop programs using decision-making and looping constructs to solve logical problems.	L3 (Apply)
CO3	Analyse and manipulate data using strings, lists, tuples, and sets.	L4 (Analyze)
CO4	Design modular programs using functions, recursion, and dictionaries.	L4 (Analyze)
CO5	Implement file handling, exception handling, and object-oriented concepts in Python applications.	L3 (Apply)

I B.Tech I Semester

Course Code	HEALTH & WELLNESS, YOGA AND SPORTS	L	T	Pr	P	Credits
26BSP12		0	0	0	1	0.5

Course Objectives

- Create awareness about holistic health, wellness and preventive healthcare among students.
- Train students in yogic practices (asanas, pranayama, meditation) for physical and mental well-being.
- Enable students to design evidence-based personal fitness and nutrition plans.
- Develop sportsmanship, team spirit, fair play and physical fitness through indoor and outdoor sports.
- Integrate technology and AI tools for health monitoring, performance analysis and wellness management.

Course Outcomes (COs)

CO No.	Course Outcome	Bloom's Level
CO1	Explain the fundamental concepts of health, wellness, fitness and the dimensions of holistic well-being.	Remember/Understand
CO2	Demonstrate correct yogic practices – asanas, pranayama and meditation – for physical and mental health.	Apply
CO3	Design a personal fitness plan integrating nutrition, exercise, stress management and lifestyle modification.	Analyse/Apply
CO4	Understand rules, skills and safety norms for a chosen sport and participate actively with sportsmanship.	Apply
CO5	Evaluate personal health indicators and reflect on lifestyle choices using technology and evidence-based practices.	Evaluate

Bloom's Taxonomy Levels: Remember (L1) | Understand (L2) | Apply (L3) | Analyse (L4) | Evaluate (L5) | Create (L6)

Unit-wise Syllabus with Learning Outcomes

UNIT 1	Health, Wellness and Preventive Healthcare	5 Hrs
Topics	• Concept of health, wellness and fitness – definitions and seven dimensions • WHO definition of health; holistic health model (Physical, Mental, Social, Spiritual) • Nutrition basics – macronutrients, micronutrients, balanced diet, BMI calculation • Lifestyle diseases – diabetes, hypertension, obesity – causes and prevention • Mental health & stress management – causes, symptoms, COPE model • Sleep hygiene and circadian rhythm regulation • Substance abuse – awareness, prevention and rehabilitation • Personal hygiene, sanitation and communicable disease prevention • Ergonomics and occupational health for students	
Unit-wise Learning Outcomes	**On completion of this unit, students will be able to:** • ULO1.1: Define health, wellness and fitness and describe the seven	

(ULOs)	dimensions of holistic well-being. [Remember] • ULO1.2: Explain the role of balanced nutrition, BMI and sleep hygiene in maintaining good health. [Understand] • ULO1.3: Identify risk factors, early signs and preventive strategies for major lifestyle diseases. [Understand] • ULO1.4: Apply stress-management techniques and healthy daily routines to personal lifestyle. [Apply] • ULO1.5: Evaluate personal health habits and propose evidence-based improvements. [Evaluate]

UNIT 2	**Yoga, Pranayama, Meditation and Mind-Body Practice**	**5 Hrs**
Topics	• Introduction to Yoga – history, philosophy, Patanjali's Ashtanga Yoga • Surya Namaskar – 12-step sequence with correct breathing • Asanas for flexibility: Tadasana, Trikonasana, Paschimottanasana, Sarvangasana • Asanas for strength & balance: Virabhadrasana I & II, Vrikshasana, Naukasana • Pranayama techniques: Anulom-Vilom, Bhastrika, Kapalbhati, Bhramari • Meditation and mindfulness – Dharana, Dhyana, body-scan, guided relaxation • Yoga Nidra for sleep and stress relief • Yoga for specific conditions – stress, back pain, PCOD/PCOS, anxiety • Shatkarmas (cleansing processes) – an overview; International Day of Yoga	
Unit-wise Learning Outcomes (ULOs)	**On completion of this unit, students will be able to:** • ULO2.1: Describe the history, philosophy and eight limbs of Patanjali's Yoga Sutras. [Remember/Understand] • ULO2.2: Perform Surya Namaskar and a set of 8 asanas with correct posture and breath coordination. [Apply] • ULO2.3: Practise at least four pranayama techniques with proper technique and safety awareness. [Apply] • ULO2.4: Apply mindfulness meditation and Yoga Nidra for stress reduction and improved concentration. [Apply] • ULO2.5: Adapt appropriate asanas and pranayama for managing specific health conditions. [Analyse]	

UNIT 3	**Physical Fitness, Sports Skills and Sports Technology**	**5 Hrs**
Topics	• Physical fitness components – strength, endurance, flexibility, agility, speed, coordination • Principles of training • – overload, progression, specificity, reversibility, individuality • Warm-up, cool-down and dynamic stretching routines • Indoor sports: Badminton, Table Tennis, Chess, Carrom – rules and basic skills • Outdoor sports: Cricket, Football, Volleyball, Kabaddi, Athletics – rules and skills • Sports injuries – types, RICE method, prevention and return-to-play	

	protocol • Sports psychology – motivation, goal setting, concentration and visualisation • Gender equity and inclusivity in sports; Para-sports awareness • Sports technology – wearables, fitness trackers, GPS analytics, performance dashboards
Unit-wise Learning Outcomes (ULOs)	**On completion of this unit, students will be able to:** • ULO3.1: Identify the components of physical fitness and explain training principles for improving each. [Understand] • ULO3.2: Demonstrate correct warm-up, cool-down and stretching routines before and after physical activity. [Apply] • ULO3.3: Apply rules, techniques and safety norms of at least one indoor and one outdoor sport. [Apply] • ULO3.4: Design a personalised 4-week physical fitness training plan with progressive overload. [Create] • ULO3.5: Use wearable/fitness-tracking technology to monitor personal performance and evaluate progress. [Evaluate]

Text Books

S.No	Title	Author(s)	Publisher	Year
1	Health and Physical Education	Dr. S. K. Mangal & Shubhra Mangal	PHI Learning Pvt. Ltd.	2021
2	The Science of Yoga	William J. Broad	Simon & Schuster	2012

Reference Books

S.No	Title	Author(s)	Publisher	Year
1	Yoga: The Iyengar Way	Silva, Mira & Shyam Mehta	Dorling Kindersley	2019
2	Sports Medicine Essentials	Jim Clover	Cengage Learning	2020
3	Foundations of Physical Education & Sport	Charles A. Bucher	McGraw-Hill Education	2018
4	Mindfulness: An Eight-Week Plan for Finding Peace	Mark Williams & Danny Penman	Rodale Books	2011

Web Resources

- Ministry of AYUSH – https://main.ayush.gov.in
- Yoga for Youth (International Day of Yoga) – https://yoga.ayush.gov.in
- National Health Portal – https://nhp.gov.in
- ICMR Dietary Guidelines – https://www.icmr.nic.in
- Sports Authority of India – https://www.sportsauthorityofindia.nic.in
- WHO Physical Activity Guidelines – https://www.who.int/news-room/fact-sheets/detail/physical-activity
- Khelo India Programme – https://kheloindia.gov.in
- NIMHANS (Mental Health Resources) – https://nimhans.ac.in

YouTube Resources

- Surya Namaskar – Complete 12-Step Tutorial – https://youtu.be/surya_tut_01
- Pranayama for Beginners (Swami Ramdev) – https://youtu.be/pranayama_sr_02
- Mindfulness Meditation (10 Min) – https://youtu.be/mindful_med_03
- Balanced Nutrition Explained (ICMR) – https://youtu.be/nutrition_icmr_04
- Sports Injury Prevention Tips – https://youtu.be/injury_prev_05
- Mental Health Awareness for Students – https://youtu.be/mh_students_06
- Yoga for Stress Relief – Isha Foundation – https://youtu.be/yoga_isha_07
- Fit India Movement – PM Modi Launch – https://youtu.be/fitindia_pmo_08

AI-Integrated Activities

S.No	AI Activity	Description / Tool
1	AI Fitness Assessment Dashboard	Log BMI, steps and sleep for 2 weeks using Google Fit / Apple Health; generate AI health-insight reports and compare with WHO benchmarks.
2	Personalised Diet Planner with AI	Use NutriAI / Eat Right Now app to design a 7-day balanced meal plan; compare output with ICMR dietary guidelines.
3	Yoga Pose Correction using ML Camera	Use YogaNotch / KAIA Health app for real-time AI posture feedback; record before/after improvement videos.
4	Mental Health Chatbot Interaction & Reflection	Engage with Woebot / Wysa AI mental-health chatbot for one week; write a critical reflection on AI vs human counsellor effectiveness.
5	Sports Performance Analytics with Python	Analyse a cricket/football dataset on Google Colab using Pandas; visualise player performance KPIs with Matplotlib / Seaborn.

I B.Tech I Semester

Course Code	COMMUNICATIVE ENGLISH LAB	L	T	Pr	P	Credits
26BSP01	Common to All Branches	0	0	0	2	1

1. COURSE OBJECTIVES

- To develop students' oral communication skills through structured speaking and listening exercises.
- To enhance pronunciation, intonation, and fluency using AI-assisted speech tools.
- To build professional communication competencies including group discussion, debate, and presentation skills.
- To enable students to write effective professional documents such as emails, reports, and cover letters.
- To integrate AI tools in language learning for self-assessment, feedback, and continuous improvement.

2. COURSE OUTCOMES

CO	Course Outcome Statement	BTL
CO1	Pronounce English words correctly using standard phonemic symbols and demonstrate improved speech fluency.	Apply
CO2	Comprehend spoken English through active listening and respond appropriately in formal and informal contexts.	Understand
CO3	Participate effectively in group discussions, presentations, and debates demonstrating logical argumentation.	Analyse
CO4	Write grammatically correct and contextually appropriate professional documents including emails, reports, and resumes.	Create
CO5	Utilize AI-powered tools for language learning, self-evaluation, and continuous improvement of English skills.	Evaluate

Note: BTL = Bloom's Taxonomy Level (Revised) | 3 = High 2 = Medium 1 = Low correlation

3. LAB SYLLABUS – 12 TOPICS WITH AI INTEGRATION

S.No	Lab Topic	Lab Activities	AI Activity	AI Tools	COs
1	**Phonetics & Pronunciation Practice**	Record & playback vowel/consonant articulation; IPA chart drill; Minimal pair exercises	Use AI speech-to-text to detect mispronunciations; real-time phoneme	ELSA Speak, Google Speech	CO1, CO5

			feedback via ELSA Speak	Recognition, Speechling	
2	**Listening Comprehension &Note-Taking**	Listen to TED Talks/podcasts; fill-in-the-blank exercises; summarise audio passages	Auto-generate transcripts; compare student notes with AI-generated summaries using Otter.ai	Otter.ai, YouTube Auto-captions, Whisper AI	**CO2, CO5**
3	**Spoken English & Fluency Building**	Impromptu speaking (1–2 min); picture description; story completion tasks	AI analyses speech for filler words, pace, and clarity; ChatGPT prompts for spontaneous speaking	Speeko, ChatGPT, Google Assistant	**CO1, CO3, CO5**
4	**Group Discussion Techniques**	Role-based GD on current topics; turn-taking practice; consensus building	AI evaluates argument structure and language; ChatGPT generates counter-arguments for practice	ChatGPT, Gemini, Mentimeter	**CO2, CO3**
5	**Public Speaking & Presentations**	Prepare and deliver 3-minute presentations; use of visual aids; Q&A handling	AI slide deck generation; real-time delivery feedback on pacing and confidence	Beautiful.ai, Gamma.app, Microsoft Copilot	**CO3, CO5**
6	**Interview Skills & Mock Interviews**	HR round simulation; STAR method answers; body language and eye contact	AI-powered mock interview with instant feedback on content, grammar, and tone	Yoodli, Interview Warmup by Google, ChatGPT	**CO1, CO3, CO5**
7	**Professional Email & Business Writing**	Draft formal emails, complaints, and requests; subject line formulation; netiquette	AI refines email drafts; tone & clarity check; compare student draft with AI-improved version	Grammarly, ChatGPT, Gemini	**CO4, CO5**
8	**Resume & Cover Letter Writing**	ATS-friendly resume design; action verbs usage; tailoring cover letters to job descriptions	AI resume scanner; keyword optimisation suggestions; ChatGPT cover letter generation and critique	Kickresume, Resume.io, Jobscan, ChatGPT	**CO4, CO5**

9	**Reading Comprehension & Critical Thinking**	Read articles/editorials; identify main idea, inference, and tone; speed reading drills	AI generates comprehension questions; text summarisation and comparative analysis	Quillbot, Speechify, NotebookLM	**CO2, CO4**
10	**Debate & Argumentation Skills**	Structured debate (for/against); Rebuttal techniques; logical fallacy identification	AI generates arguments on both sides; evaluate logical structure using ChatGPT	ChatGPT, Claude.ai, Kialo Edu	**CO2, CO3**
11	**Report Writing & Technical Communication**	Lab report, incident report, and project report formats; headings, data presentation	AI improves report coherence and passive/active voice usage; grammar and structure feedback	Grammarly Business, ChatGPT, Hemingway Editor	**CO4, CO5**
12	**Etiquette, Netiquette &Personal Branding**	Workplace conversation role play; LinkedIn profile creation; professional social media etiquette	AI reviews LinkedIn summaries; digital persona analysis; personal brand statement via ChatGPT	ChatGPT, Canva AI, LinkedIn AI features	**CO3, CO4, CO5**

4. TEXT BOOKS

S.No	Reference
1	Raman, Meenakshi& Sharma, Sangeeta. Technical Communication: Principles and Practice (3rd Ed.). Oxford University Press, 2022.
2	Krishnaswamy, N. &Krishnaswamy, L. The Story of English in India. Foundation Books, Cambridge University Press, 2021.

5. REFERENCE BOOKS

S.No	Reference
1	Swan, Michael. Practical English Usage (4th Ed.). Oxford University Press, 2016.
2	Murphy, Raymond. English Grammar in Use (5th Ed.). Cambridge University Press, 2019.
3	Rizvi, M. Ashraf. Effective Technical Communication (2nd Ed.). McGraw-Hill

	Education, 2018.
4	Turton, N.D. & Heaton, J.B. Longman Dictionary of Common Errors. Pearson Longman, 2017.
5	Lesikar, Raymond V. &Flatley, Marie E. Basic Business Communication (12th Ed.). McGraw-Hill, 2020.
6	Rutherford, Andrea J. Basic Communication Skills for Technology (3rd Ed.). Pearson, 2019.

6. ONLINE RESOURCES

- BBC Learning English – https://www.bbc.co.uk/learningenglish (Grammar, pronunciation, and listening)
- British Council Learn English – https://learnenglish.britishcouncil.org
- TED Talks – https://www.ted.com/talks (Listening & speaking practice)
- Coursera English Communication Skills Specialisation – https://www.coursera.org
- edX English for Professional and Academic Purposes – https://www.edx.org
- Grammarly Blog – https://www.grammarly.com/blog (Writing tips)
- ELSA Speak (App) – https://elsaspeak.com (AI pronunciation coach)

7. YOUTUBE CHANNELS

- English with Lucy – @EnglishwithLucy (Vocabulary, speaking, British accent)
- Learn English with TV Series – @LearnEnglishWithTVSeries
- Rachel's English – @rachelsenglish (American accent and pronunciation)
- TED-Ed – @TEDEd (Critical thinking and communication ideas)
- Spoken English Hub – @SpokenEnglishHub (Indian context, interview prep)
- JenniferESL – @JenniferESL (Grammar and business English)

8. NPTEL / SWAYAM COURSES

Course Name	Platform	Provider	Link
Developing Soft Skills and Personality	NPTEL / SWAYAM	IIT Kanpur	https://swayam.gov.in
Business English Communication Suite	Coursera (Free Audit)	University of Washington	https://coursera.org
Speak English Professionally: In Person, Online & On the Phone	Coursera	Georgia Tech	https://coursera.org

English and Academic Preparation	NPTEL	IIT Bombay	https://nptel.ac.in
Technical English for Engineers	NPTEL / SWAYAM	IIT Roorkee	https://swayam.gov.in

9. LAB SOFTWARE & AI TOOLS REQUIRED

Software / Tool	Type	Purpose
ELSA Speak	AI Pronunciation Coach App	Real-time speech accuracy & fluency feedback
Grammarly	AI Writing Assistant	Grammar, punctuation, style correction
ChatGPT / Claude.ai	Generative AI Chatbot	Conversation practice, writing assistance, debate prep
Otter.ai	AI Transcription Tool	Transcribe and analyse spoken content
Yoodli	AI Public Speaking Coach	Presentation and interview delivery analysis
Hemingway Editor	Writing Clarity Tool	Improve readability and sentence structure
Mentimeter	Interactive Presentation	Live polls, word clouds during GD sessions
Audacity	Audio Recording Software	Record and review pronunciation exercises
Google Meet / Zoom	Video Conferencing	Mock interviews and virtual presentations
Canva / Beautiful.ai	AI Presentation Design	Create professional visual presentations

I B.Tech II Semester

Course Code	AI TOOLS AND APPLICATIONS	L	T	Pr	P	Credits
26AIP01	Common to All Branches	1	0	4	0	3
Prerequisite:	None					

Course Objectives

1. Introduce students to AI concepts and the landscape of AI tools available for every day academic and professional use.
2. Develop practical skills in using AI for document creation, presentations, writing, and academic tasks applicable across all engineering disciplines.
3. Build proficiency in AI-powered data analysis, spreadsheet automation, and intelligent research tools without requiring programming knowledge.
4. Expose students to AI tools in visual design, media creation, and note-taking,and progressively to AI applications specific to their engineering branch.
5. Cultivate responsible AI habits: verifying outputs, protecting privacy, respecting intellectual property, and understanding the ethical and legal boundaries of AI use.

Course Outcomes(COs)

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

CO	Course Outcome Statement	BTL
CO1	Explain what AI is, how it works conceptually, and describe the landscape of AI tools across text, image, audio, video, and code categories.	L2
CO2	Use AI tools to create documents and presentations; apply prompt engineering and Note book LM to improve writing, research, and study habits.	L3
CO3	Perform data analysis using AI without coding; use AI-powered research and note-taking tools to find, verify, organize, and summarize information.	L3
CO4	Apply AI tools in visual design, diagrams, and media; identify and verify AI applications relevant to their own engineering discipline.	L4
CO5	Demonstrate responsible AI usage — bias, privacy, ethics, governance; design and present an AI-assisted mini-project using tools from all units.	L5

Course Content

UNIT–I	UNDERSTANDING AI AND AIFORDOCUMENTS AND PRESENTATIONS	9 Hrs

- **What is AI:** AI vs. ML vs. Deep Learning vs. Generative AI; brief history (Turing Test → ChatGPT → generative AI era); how LLMs work (training data, next-token prediction, temperature, hallucinations, context windows, knowledge cutoffs). AI ecosystem: ChatGPT, Claude, Gemini, Microsoft Copilot — strengths and differences.
- **AI for presentations:** Gamma.app — full slide deck from a paragraph prompt; Microsoft CopilotinPowerPoint—slidegeneration,designsuggestions,speakernotes;CanvaAI—

Magic Design, text-to-presentation, brand kits; Beautiful.ai — smart slide layouts; comparing AI-generated vs. manually created presentations.

- **AI for document writing:** Microsoft Co pilotin Word —draft, rewrite, summarize; Geminiin Google Docs — writing assistance and inline Q&A; Notion AI — meeting notes, project plans, knowledge base; Claude Projects — organizing documents and a persistent AI assistant over a set of files.
- **AI writing assistance:** Grammarly — grammar, tone, clarity, plagiarism; ChatGPT and Claude for lab reports, proposals, and letters; academic integrity — AI as an assistant not an author; disclosure norms; identifying AI-generated content using GPT Zero.

AI Integration: *Use Gamma.app to generate a presentation on an engineering topic in under 2 minutes; use Copilot in Word or Gemini in Docs to draft and improve a report; compare ChatGPT, Claude, Gemini, and Microsoft Copilot on the same writing task; use Grammarly to review your own writing.*

UNIT–II	AI FOR RESEARCH, NOTE-TAKING, DATA, AND SPREADSHEETS	9 Hrs

- **Notebook LM (Google):** Uploading lecture notes, textbooks, PDFs, and research papers as sources; asking questions grounded in uploaded documents; generating study guides, timelines, briefing documents, and audio overviews (podcast-style summaries); source-grounded answers that cite the exact passage — eliminating hallucination risk.
- **AI for research:** Perplexity AI (real-time web search with citations), Gemini Deep Research (multi-step research reports), Elicit and Consensus (academic paper search with evidence summaries); verifying AI-generated citations on Google Scholar; why AI fabricates references and how to detect it.
- **Prompt engineering:** Zero-shot, one-shot, few-shot, chain-of-thought, role-based prompting; iterative refinement; negative prompting; building effective prompts for summarization, comparison, explanation, and data extraction; using ChatGPT custom instructions and Claude Projects for persistent context.
- **AI for spreadsheets and data analysis:** Microsoft Copilot in Excel — VLOOKUP, IF, COUNTIF, pivot tables from natural language; Gemini in Google Sheets — formula generation and chart creation; ChatGPT Advanced Data Analysis and Julius AI — uploading CSV files, computing statistics, generating charts, interpreting patterns — no coding required.

AI Integration: *Upload your own lecture notes to Notebook LM and generate a study guide and audio overview; use Perplexity AI and Gemini Deep Research to research a topic; verify citations on Google Scholar; use Wolfram Alpha or Symbol ab to solve and verify equations from your current subjects; use Copilot in Excel or ChatGPT ADA to analyse a real dataset.*

UNIT–III	AI FOR VISUAL DESIGN,DIAGRAMS,AND MEDIA	9 Hrs

- **AI for image generation:** DALL-E 3 (via ChatGPT), Canva AI (text-to-image, Magic Design, background remover), Adobe Firefly, Ideogram — writing effective image prompts (style, subject, composition, colour); iterative prompt refinement for posters, diagrams, and mockups.
- **AI for diagrams and visual thinking:** Napkin.ai — converting text or concepts into visual diagrams, charts, and in fographics automatically; Whimsical AI — AI-generated mind maps, flowcharts, and wireframes from a description; Miro AI — collaborative whiteboard with AI summarization and diagramming.
- **AI for audio and video:** ElevenLabs — text-to-speech and voice cloning for presentations and voiceovers; Suno and Udio — AI music generation for project videos; Runway and Pika —

generating short video clips from text prompts; HeyGen — AI avatars for video presentations; current limitations and appropriate use cases.

- **AI content detection and critical evaluation:** GPTZero, Originality.ai for AI-text detection; Google Lens and reverse image search for AI-generated images; fact-checking workflows — cross-referencing primary sources; deepfakes and synthetic media — recognition, risks, and responsible creation.

AIIntegration: *UseNapkin.ai to convert a paragraph of text into a diagram;use Whimsical AI to generate a mind map of a course topic; use DALL-E 3 or Canva AI to create an event poster through 3 prompt iterations; use ElevenLabs to create a voiceover; test GPTZero on 5 samplesand analyse confidence scores.*

UNIT–IV	AI ACROSS ENGINEERING DISCIPLINES	9 Hrs

- **AI in Mechanical Engineering:** Generative design in CAD (Autodesk Fusion 360 — AI-optimized structures for weight and load);predictive maintenance using sensor data and ML;AI-assisted FEA simulation; manufacturing process optimization; digital twins.
- **AI in Civil and Electrical Engineering:** Structural health monitoring using ML; smart grid load forecasting and energy management; AI for circuit simulation and PCB layout optimization (Altium AI); VLSI design automation — floor planning and routing; power systems fault detection.
- **AI in Electronics and Communication:** AI-assisted signal processing and filtering; speech recognition for embedded systems; computer vision for industrial quality inspection; AI in antenna design and spectrum management; MATLAB AI and Deep Learning Toolbox.
- **AI in Computer Science and allied branches:** GitHub Copilot for code generation, explanation, and review; AI-powered testing (Testim, Selenium AI); AI in cyber security — anomaly and intrusion detection; no-code/low-code platforms (Bubble, Glide, Webflow AI); AI for data science (Jupyter AI, DataRobot).

AI Integration: *Each student identifies 5 real AI tools in their own engineering branch using Claude or ChatGPT and verifies each using official product pages or research publications; present findings in a 3-minute class presentation with a structured comparison table.*

UNIT–V	RESPONSIBLE AI, ETHICS, AND CAPSTONE PROJECT	9 Hrs

- **AI bias and fairness:** Sources of bias (data, model, deployment); real documented cases — Amazon hiring algorithm (2018), COMPAS recidivism (2016), facial recognition disparities (NIST 2019); fairness metrics; misinformation, deepfakes, and synthetic media risks.
- **Privacy and intellectual property:** What happens to data entered into public AI tools; PII risks; data retention policies of ChatGPT (OpenAI), Claude (Anthropic), and Gemini (Google); who owns AI-generated content; copyright status globally; academic plagiarism with AI; personal data safety protocol.
- **AI governance frameworks:** EU AI Act risk tiers (unacceptable, high, limited, minimal); UNESCO AI Ethics Recommendation; India's NITI Aayog Responsible AI Guidelines; India Digital Personal Data Protection Act 2023 (DPDP Act); environmental cost of AI — energy and carbon footprint of LLM training and inference.
- **Capstone Mini-Project:** Each student (or pair) defines a real engineering problem, selects at least 3 AI tool categories from Units I–IV, produces a quality-verified deliverable (report, presentation, diagram, or automation), and presents in 5 minutes. Evaluation: problem clarity, tool selection, output quality, human refinement, responsible AI disclosure.

AI Integration: *Read and compare data retention policies of ChatGPT, Claude, and Gemini; classify 5daily-useAIsystemsusingEUAIActrisktiers;use GPT Zero and Google Lens to detect AI-generated content; design and present the Capstone Mini-Project integrating tools from all five units.*

Learning Resources Reference Books

1. Ethan Mollick, Co-Intelligence: Living and Working with AI,Portfolio/Penguin,2024.
2. Stuart J.Russell and Peter Norvig, Artificial Intelligence: A Modern Approach,4th Edition, Pearson Education, 2020.
3. Reid Blackman, Ethical Machines,Harvard Business Review Press, 2022.

Online Learning

1. Google–Fundamentals of AI(grow.google)•Microsoft–AISkillsNavigator (microsoft.com/ai/ai-skills)
2. Andrew Ng–AI for Everyone,Coursera,free to audit(coursera.org/learn/ai-for-everyone)
3. Elements of AI–University of Helsinki, free, beginner (elementsofai.com)

AI Tools by Category

1. **AI Assistants:**ChatGPT,Claude,Gemini,MicrosoftCopilot
2. **Research &Notes:**Note book LM,PerplexityAI, Elicit,Consensus
3. **Presentations & Docs:** Gamma.app, Notion AI, Beautiful.ai, Co pilotin Office365, Gemini in Google Workspace
4. **Diagrams&Design:**Napkin.ai,WhimsicalAI,MiroAI,CanvaAI,DALL-E3via ChatGPT
5. **Data & Spreadsheets:** ChatGPT Advanced Data Analysis, Julius AI, Copilot in Excel, Gemini in Sheets
6. **Math&Equations:**WolframAlpha,Symbolab,Mathway,MicrosoftMathSolver, Photomath, Desmos
7. **Audio &Video:** Eleven Labs, Suno, Runway, HeyGen
8. **Detection&Automation:**GPTZero,Originality.ai,GoogleLens,Make.com,GitHub Copilot
9. **Governance&Ethics:**EUAIAct(eur-lex.europa.eu),UNESCOAIEthics,NITIAayog, India DPDP Act 2023, ProPublica COMPAS

LIST OF EXPERIMENTS

30 Experiments (6 per Unit)•2 hours each •25 marks per experiment

UNIT–I•Understanding AI and AI for Documents and Presentations*(6Programs)*

No.	Tool/Experiment Title
1	ChatGPT& Claude—Understanding LLMs, temperature, and hallucination
2	Gamma.app—AI-generated presentation from a paragraph prompt
3	Microsoft Co pilotin Word—draft, rewrite,and summarize
4	Grammarly—grammar, tone,clarity,and plagiarism detection
5	GPTZero—AI text detection: test 5 samples and analyze scores
6	Google Gemini— multimodal prompting with image and text input

UNIT–II•AI forResearch,Note-taking,Data,and Spread sheets*(6 Programs)*

No.	Tool/Experiment Title
1	Note book LM—upload lecture notes;generate study guide+audio overview
2	Perplexity AI—real-time research with verified citations
3	Elicit &Consensus—academic papersearch and evidence synthesis

4	Prompt engineering—zero-shotto chain-of-though to engineering tasks
5	Copilotin Excel/Julius AI— natural-language data analysis of a CSV data set
6	Zotero+Sci Space— reference management and AI paper Q&A

UNIT–III•AI forVisual Design, Diagrams,and Media*(6 Programs)*

No.	Tool/Experiment Title
1	DALL-E3/CanvaAI—eventposterthrough3iterativeprompt refinements
2	Napkin.ai & Whimsical AI— concept diagram and course mind map
3	Miro AI—collaborative system architecture brain storm
4	Eleven Labs—create 60-second technical project voice over
5	Google Lens & GPT Zero— detect AI-generated text and images
6	Text-to-video—30-second technical explainer with CanvaAI/ Runway

UNIT–IV•AI Across Engineering Disciplines*(6 Programs)*

No.	Tool/Experiment Title
1	Git Hub Copilot—code generation, explanation,and review
2	ChatGPT ADA—full data science pipe line on areal dataset
3	Bubble.ioor Glide— buildano- codeAI-powered app proto type
4	Wolfram Alpha & Symbo lab—solve and verify engineering equations
5	Branch AI tool research—identify and verify 5real branch-specific AI tools
6	Teachable Machine—train and deploya no-code image classifier

UNIT–V•ResponsibleAI,Ethics,and Capstone Project*(6 Programs)*

No.	Tool/Experiment Title

No.	Tool/ExperimentTitle
1	AI bias audit—analyze COMPAS fairness metrics
2	Privacy audit—compare data retention policies of ChatGPT,Claude,Gemini
3	EU AI Act classification—classify 5daily-use AI systems
4	AI hallucination & fact-checking audit — a responsible verification workflow
5	Capstone deliverable—integrated multi-tool engineering solution
6	Capstone presentation—5-minute project demo with responsible AI disclosure

Note: Free / open-source tiers of the listed tools are sufficient for all experiments. Each experiment is recorded in the lab journal with Aim, Requirements, Procedure, Input, Output/Screenshots, Observations, Result, and Viva answers.

Bloom's Taxonomy Coverage–Summary

Bloom's Level	Level Name	CO(s)	Sample Activities/Assessments
L1–L2	Remember/ Understand	**CO1**	Define AI, GenAI, LLM; describe prompt types and tool categories.
L3	Apply	**CO1–CO3**	Use AI tools to create presentations, research summaries, and visual diagrams.
L4	Analyse	**CO2, CO3**	Evaluate prompt effectiveness; compare tools for a given task.
L5	Evaluate	**CO4, CO5**	Assess tool appropriateness per discipline; evaluate ethical implications.
L6	Create	**CO1, CO5**	Design an AI-assisted project deliverable with responsible usage documentation.

CO–PO/PSOArticulationMatrix(NBA)

Scale:3 –Strong|2–Moderate|1 – Slight|– No Correlation

CO	Bloom's Lvl	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	L3	**3**	**2**	**2**	1	**3**	**2**	–	1	1	**2**	1
CO2	L4	**3**	**3**	**2**	**2**	**3**	1	–	1	1	**2**	1
CO3	L4	**3**	**2**	**3**	**2**	**3**	1	–	1	1	**2**	1
CO4	L5	**3**	**3**	**3**	**3**	**3**	**2**	–	1	**2**	**2**	**2**
CO5	L5	**3**	**2**	**2**	**2**	**3**	**2**	1	**2**	**2**	**2**	**2**
CO-PO Correlation Highlights	Strong PO5 (Modern Tools) through Gamma.app, NotebookLM, ChatGPT, Claude, Napkin.ai, and domain-specific AI tools across all units. PO3 (Design/Development) through CO3 (visual design and diagram creation). PO6 (Engineer & Society) and PO8 (Ethics) through responsible AI usage, bias awareness, and the capstone ethics module in CO5.											

ReferenceBooks

1. Ethan Mollick, Co-Intelligence: Living and Working with AI, Portfolio/Penguin, 2024.
2. Stuart J. Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, Pearson Education, 2020.
3. Reid Blackman, Ethical Machines, Harvard Business Review Press, 2022.
4. Google – Fundamentals of AI – grow.google/intl/en_in/ai-learning-resources
5. Microsoft – AI Skills Navigator – microsoft.com/en-us/ai/ai-skills
6. Andrew Ng – AI for Everyone (Coursera, free to audit) – coursera.org/learn/ai-for-everyone
7. Elements of AI – University of Helsinki (free, beginner) – elementsofai.com
8. AI Assistants: ChatGPT – chat.openai.com | Claude – claude.ai | Gemini – gemini.google.com | Microsoft Copilot – copilot.microsoft.com
9. Research & Notes: NotebookLM – notebooklm.google.com | Perplexity AI – perplexity.ai | Elicit – elicit.org | Consensus – consensus.app
10. Presentations & Docs: Gamma.app | Notion AI – notion.so | Beautiful.ai | Copilot in Office 365 | Gemini in Google Workspace
11. Diagrams & Design: Napkin.ai | Whimsical AI – whimsical.com | Miro AI – miro.com | Canva AI – canva.com | DALL-E 3 via ChatGPT
12. Data & Spreadsheets: ChatGPT Advanced Data Analysis | Julius AI – julius.ai | Copilot in Excel | Gemini in Sheets
13. Math & Equations: Wolfram Alpha – wolframalpha.com | Symbolab – symbolab.com | Mathway – mathway.com | Microsoft Math Solver – math.microsoft.com | Photomath – photomath.com | Desmos – desmos.com | GeoGebra – geogebra.org
14. Audio & Video: ElevenLabs – elevenlabs.io | Suno – suno.com | Runway – runwayml.com | HeyGen – heygen.com
15. Detection & Automation: GPTZero – gptzero.me | Originality.ai | Google Lens | Make.com | GitHub Copilot

Online Resources

- https://www.youtube.com/@JoyofLearningCSE/featured
- EU AI Act – eur-lex.europa.eu | UNESCO AI Ethics – unesco.org | NITI Aayog – niti.gov.in

India DPDP Act 2023 – meity.gov.in | ProPublica COMPAS – propublica.org | MIT Technology Review – technologyreview.com |

I B.Tech II Semester

Course Code	PHYSICS FOR ADVANCED COMPUTING	L	T	Pr	P	Credits
26BST13	Common to CSE & Allied Branches	3	0	0	0	3

Prerequisites: Engineering Mathematics, Basic Electronics, Computer Science Fundamentals

Course Objectives

- To equip students with the concepts of interference, diffraction and polarization and their applications in optics and communication systems with basic AI-assisted analysis.
- To introduce lasers, fiber optics and advanced sensing technologies along with simple AI-assisted signal processing techniques.
- To provide basic knowledge of quantum mechanics and quantum computing with simple AI-assisted quantum applications.
- To familiarize students with semiconductors and magnetic materials and their applications using basic AI-assisted material analysis methods.
- To impart knowledge on low temperature physics and superconductivity and their applications with simple AI-assisted monitoring techniques.

Course Outcomes (COs)

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

CO	Course Outcome Statement	Bloom's Level
CO1	Apply the principles of interference, diffraction and polarization to analyze optical systems and engineering applications.	**L3 (Apply)**
CO2	Analyze the working of lasers, fiber optics and sensing systems with basic AI-assisted optical communication concepts.	**L4 (Analyze)**
CO3	Apply the concepts of quantum mechanics and quantum computing to explain modern computational and secure communication systems.	**L3 (Apply)**
CO4	Analyse semiconductor characteristics and magnetic material behaviour for engineering and electronic applications.	**L4 (Analyze)**
CO5	Apply the concepts of low-temperature physics and superconductivity in modern technological applications and intelligent systems.	**L3 (Apply)**

Course Content

UNIT – 1	Wave Optics and Quantum Information Processing	9 Hrs

- Interference: Principle of superposition and coherence; Interference; Types of interference; Interference in thin films (reflection geometry) and Newton's rings. Applications: Determination of wavelength and refractive index.
- Diffraction: Introduction to diffraction; Fresnel and Fraunhofer diffraction; Fraunhofer diffraction due to single, double and N-slits (Diffraction Grating); Dispersive power and Resolving power of grating. Applications: CD/DVD data storage, Barcode scanners.
- Polarization: Types of polarization; Nicol's prism; Half-wave and Quarter-wave plates.

AI-assisted polarization management for quantum key distribution. AI-integrated holographic interferometry for surface flatness analysis. Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib and scikit-learn).

UNIT – 2	Quantum Photonics and Advanced Sensing	9 Hrs

- Lasers: Interaction of radiation with matter; Requisites and conditions for laser action; He-Ne laser; Nd-YAG laser; LiDAR basics. Laser-based data communication, Free-space optical communication, Optical data storage, Photonic computing basics, Applications of lasers.
- Fiber Optics: Propagation mechanism; Numerical aperture; Acceptance angle; Types of optical fibres; Classifications and Modes of optical fibres; Attenuation and Losses in optical fibers; V-Number; Distributed Fiber Optic Sensing (DFOS) with AI-Powered Signal Analysis.

AI-powered signal analysis for distributed fiber optic sensing. Python simulation of laser propagation and fiber optic signal attenuation. Free/Open-Source AI Tools: Google Colab, Python, NumPy, SciPy and Scikit-learn/TensorFlow.

UNIT – 3	Introduction to Quantum Mechanics and Quantum Computing	10 Hrs

- Introduction to quantum mechanics; de-Broglie concept (matter waves); Heisenberg's Uncertainty Principle; Wave function; Postulates of Quantum Mechanics; Schrödinger's time-independent wave equation; Particle in a one-dimensional potential well; Quantum tunnelling (basic idea).
- Introduction to Quantum Computing: Classical bits vs Qubits; Concept of superposition and Entanglement; Basic Quantum Operations: Hadamard gate (idea of superposition), Pauli-X gate; Basic idea of quantum measurement.
- Applications (qualitative only): AI-Assisted Tunnel Diodes and cold emission of electrons; Basic idea of quantum cryptography and secure communication.

AI-Assisted Analysis of Tunnel Diode Characteristics using Quantum Tunnelling Concepts. AI-Assisted Quantum Key Distribution for Secure Communication. Free/Open-Source AI Tools: Google Colab, Python and Scikit-learn.

UNIT – 4	Semiconductors and Magnetic Materials	8Hrs

Semiconductors: Introduction; Fermi energy and Fermi level in intrinsic and extrinsic semiconductors; Expression for concentration of electrons in conduction band and holes in valence band; Electrical conductivity; Expressions for drift and diffusion currents; Hall effect; Expression for Hall coefficient and its applications.

Magnetism: Origin of magnetism; Dia, Para, Ferro, Ferri and Anti-ferro magnetic materials; Hysteresis; Soft and Hard magnetic materials. Applications (qualitative only): Magnetic data storage (MRAM), Hard Disk Drive and Magnetic shielding.

AI-Assisted Hall Effect Analysis for Semiconductor Characterization. AI-Assisted Magnetic Data Storage and MRAM Optimisation. Free/Open-Source AI Tools: Google Colab, Python and Scikit-learn.

UNIT – 5	Low Temperature Physics and Superconductivity	9 Hrs

Introduction: Low temperature and its importance; Properties of materials at low temperatures; Liquefaction of gases (basic concept only); Liquid nitrogen and liquid helium (basic properties and uses). Applications (basic ideas): MRI, Rocket fuels, Cryogenic Cooling of High-Performance Computing System.

Superconductors: Definitions; Meissner effect; Type I and Type II superconductors; BCS theory: Cooper pairs and energy gap (qualitative); DC and AC Josephson effect. Applications (qualitative only): SQUIDs, magnetic levitation and lossless power transmission. Superconducting qubits: Charge, Flux, Phase.
AI-Assisted Cryogenic Cooling Optimization for High-Performance Computing Systems. AI-Assisted Stability Control in Superconducting Magnetic Levitation Systems. Free/Open-Source AI Tools: Google Colab, Python and Scikit-learn.

CO – PO Articulation Matrix (NBA)

Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	1	2	-	-	-	-	-	1
CO2	3	3	2	2	2	-	-	-	-	-	1
CO3	3	2	2	2	2	-	-	-	-	-	1
CO4	3	3	2	2	2	-	-	-	-	-	1
CO5	3	2	1	2	2	-	-	-	-	-	1

LEARNING RESOURCES

Textbooks

1. Avadhanulu, M. N., Kshirsagar, P. G., & Arun Murthy, T. V. S. (2019). Engineering Physics. S. Chand.
2. Bhattacharya, D. K., & Tandon, P. (2015). Engineering Physics. Oxford Press.

Reference Books

1. Pandey, B. K., & Chaturvedi, S. (2021). Engineering Physics. Cengage Learning.
2. Pillai, S. O. (n.d.). Solid State Physics. New Age International.
3. Nielsen, M. A., & Chuang, I. L. (n.d.). Quantum Computation & Quantum Information. Cambridge University Press.
4. Saleh, B. E. A., & Teich, M. C. (n.d.). Fundamentals of Photonics. Wiley.
5. Keiser, G. (n.d.). Optical Fiber Communications. McGraw Hill.
6. Tinkham, M. (n.d.). Introduction to Superconductivity. Dover Publications.

Online Resources

- NPTEL – nptel.ac.in
- MIT OpenCourseWare – ocw.mit.edu
- HyperPhysics – hyperphysics.phy-astr.gsu.edu
- PhET Simulations – phet.colorado.edu

- IBM Qiskit (free, open-source) – For quantum computing and superconducting qubit simulations on Google Colab

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for wave optics and signal analysis
- IBM Qiskit (free, open-source) – For quantum computing and superconducting qubit simulations on Google Colab
- TensorFlow / Keras (free, open-source) – For photonic and optical pattern recognition
- scikit-learn (free, open-source) – For semiconductor property prediction and LiDAR signal classification
- PhET Interactive Simulations (free) – For wave optics, quantum mechanics and semiconductors

I Year - II Semester

Course Code	DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS (Common to all Branches)	L	T	Pr	P	Credits
26BST03		3	0	2	0	4
Prerequisite:	Linear Algebra and Calculus					

Course Objectives

- Develop analytical and computational competence in solving ordinary and partial differential equations arising in engineering systems.
- Enable students to interpret vector differential and integral operators in physical and engineering contexts.
- Integrate mathematical modeling, visualization, and simulation using modern AI-assisted computational tools in alignment with NEP 2020 and outcome-based education.
- Apply ODE and PDE solution methods to model real-world engineering phenomena such as circuits, vibrations, heat transfer, and fluid flow.
- Use computational tools for solving, verifying, and visualizing differential equations and vector field problems.

Course Outcomes (COs)

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

CO	Course Outcome Statement	Bloom's Level
CO1	Solve first-order ODEs and model engineering processes using analytical and computational approaches.	**L3 (Apply)**
CO2	Analyze higher-order linear ODEs and interpret solutions in dynamic engineering systems such as circuits and oscillators.	**L4 (Analyze)**
CO3	Formulate and solve first-order and selected higher-order PDEs relevant to engineering models including wave, heat, and Laplace equations.	**L3 (Apply)**
CO4	Interpret and compute gradient, divergence, and curl of scalar and vector fields and explain their physical significance in engineering contexts.	**L4 (Analyze)**
CO5	Apply Green's, Stokes', and Divergence theorems to compute work, circulation, and flux and evaluate engineering field behavior.	**L5 (Evaluate)**

Course Content

UNIT – 1	First-Order Ordinary Differential Equations	9 Hrs
Content	• Classification of ODEs by order, degree, and linearity; linear first-order ODEs and integrating factor method; Bernoulli's equation; exact equations and integrating factors. • Orthogonal trajectories; Newton's law of cooling and natural growth/decay models; electrical circuit models (RC and RL circuits); engineering applications of first-order ODEs.	

AI Integration	Python SymPy dsolve for symbolic ODE solution and verification; direction field (slope field) visualization using Matplotlib; AI-assisted debugging of integrating factor derivations; using ChatGPT/Copilot to explain solution families and physical interpretations; SciPy solve_ivp for numerical ODE solution. Free/Open-Source AI Tools: Python (SymPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; GNU Octave.	
UNIT – 2	**Higher-Order Linear Differential Equations**	**9 Hrs**
Content	• Superposition principle; Wronskian and linear independence; complementary function for constant-coefficient ODEs (distinct real, repeated, and complex roots); particular integral by operator method; variation of parameters. • Simultaneous linear ODEs; Cauchy's (Euler) equation; Legendre's equation; applications to LCR circuits, simple harmonic motion, damped and forced oscillations; mass-spring-dashpot systems.	
AI Integration	Python SciPy signal.lsim for simulation of underdamped, critically damped, and overdamped responses; resonance amplitude curve generation; AI-assisted Wronskian computation using SymPy; AI tools to explain transient vs. steady-state response. Free/Open-Source AI Tools: Python (NumPy, SciPy, SymPy, Matplotlib); GNU Octave; WolframAlpha free tier.	
UNIT – 3	**Partial Differential Equations**	**9 Hrs**
Content	• Introduction and classification of PDEs; formation by elimination of arbitrary constants; formation by elimination of arbitrary functions; first-order linear PDEs and Lagrange's method (auxiliary equations); homogeneous linear PDEs with constant coefficients (CF and PI). • Standard forms of nonlinear first-order PDEs; second-order PDE applications — wave equation, heat equation, and Laplace's equation; classification as hyperbolic, parabolic, and elliptic.	
AI Integration	Python SymPy for PDE formation and symbolic verification; 3D surface plot visualization of PDE solution families using Matplotlib; SciPy finite-difference simulation of the 1D heat equation with animation; AI-assisted PDE classification using the discriminant. Free/Open-Source AI Tools: Python (SymPy, NumPy, SciPy, Matplotlib); WolframAlpha free tier; Scilab.	
UNIT – 4	**Vector Differentiation**	**9 Hrs**
Content	• Scalar and vector fields; del (∇) operator; gradient and directional derivative; unit normal to surfaces; angle between surfaces; divergence and physical interpretation; curl and physical interpretation. • Vector identities; conservative fields and scalar potentials; Laplacian; engineering applications of gradient, divergence, and curl in electromagnetic and fluid mechanics contexts.	

AI Integration	Python NumPy and Matplotlib for 2D and 3D gradient, divergence, and curl field visualization (quiver plots, level-surface overlays); SymPy for symbolic verification of vector identities; AI tools to interpret Maxwell's equations in terms of divergence and curl operators. Free/Open-Source AI Tools: Python (NumPy, SymPy, Matplotlib, mpl_toolkits); SciPy for numerical computation.

UNIT – 5	**Vector Integration and Integral Theorems**	**9 Hrs**
Content	• Line integrals over scalar and vector fields; work done and path independence; surface integrals and flux; Green's theorem in the plane and its applications. • Stokes' theorem; Divergence theorem (Gauss's theorem); applications to area and volume computation; engineering applications in electromagnetism, fluid mechanics, and mass conservation; relationship between integral theorems and Maxwell's equations.	
AI Integration	Python SciPy integrate for numerical computation of line and surface integrals; 3D flux visualization with colour-coded normal flux using Matplotlib; numerical verification of Green's and Stokes' theorems; AI-assisted physical interpretation of the Divergence theorem. Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib); SymPy for symbolic integral verification; WolframAlpha free tier.	

Bloom's Taxonomy Coverage – Summary

Bloom's Level	Level Name	CO(s)	Sample Activities
L3	Apply	**CO1, CO3**	First-order ODE solution problems; PDE formulation and Lagrange's method exercises.
L4	Analyze	**CO2, CO4**	Higher-order ODE response analysis for LCR circuits; gradient/divergence/curl field interpretation.
L5	Evaluate	**CO5**	Green's/Stokes'/Divergence theorem application for flux and circulation computation.

CO – PO Articulation Matrix (NBA)

Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	2	2	1	-	-	-	1	2	-
CO2	3	3	3	2	2	-	-	-	1	2	-
CO3	3	2	2	2	1	-	-	-	-	2	-
CO4	3	3	3	3	2	-	-	-	1	2	-
CO5	3	3	2	3	3	-	1	1	2	2	1

CO-PO Correlation Highlights	CO1 through CO4 strongly align with PO1 and PO2 through ODE/PDE solving and vector field analysis. CO5 maps additionally to PO7 and PO8 through engineering field interpretation using integral theorems in electromagnetism and fluid mechanics. PO5 (Modern Tools) is addressed throughout via SymPy, SciPy, and Google Colab-based simulation across all five units, while PO12 (Lifelong Learning) is consistently reinforced through self-directed computational verification practice.

LEARNING RESOURCES

Textbooks

1. Kreyszig, E. (2018). Advanced Engineering Mathematics (10th ed.). Wiley.
2. Grewal, B. S. (2017). Higher Engineering Mathematics (44th ed.). Khanna Publishers.

Reference Books

1. Thomas, G. B., Weir, M. D., & Hass, J. (2018). Thomas' Calculus (14th ed.). Pearson.
2. Jain, R. K., & Iyengar, S. R. K. (2021). Advanced Engineering Mathematics (5th ed.). Alpha Science International.
3. Zill, D. G., & Wright, W. S. (2018). Advanced Engineering Mathematics (6th ed.). Jones & Bartlett.
4. Arfken, G. B., Weber, H. J., & Harris, F. E. (2012). Mathematical Methods for Physicists (7th ed.). Academic Press.
5. Sneddon, I. N. (2006). Elements of Partial Differential Equations. Dover Publications.
6. Davis, H. F., & Snider, A. D. (1995). Introduction to Vector Analysis (7th ed.). Hawkes Publishing.

Online Resources

- NPTEL: Ordinary Differential Equations and Applications – https://nptel.ac.in
- NPTEL: Partial Differential Equations – https://nptel.ac.in
- NPTEL: Vector Calculus – https://nptel.ac.in
- MIT OpenCourseWare – Differential Equations (18.03): https://ocw.mit.edu

1. MIT OpenCourseWare – Multivariable Calculus (18.02): https://ocw.mit.edu

Web Resources & NPTEL Links

- NPTEL: Ordinary Differential Equations and Applications – https://nptel.ac.in
- NPTEL: Partial Differential Equations – https://nptel.ac.in
- NPTEL: Vector Calculus – https://nptel.ac.in
- MIT OpenCourseWare – Differential Equations (18.03): https://ocw.mit.edu
- MIT OpenCourseWare – Multivariable Calculus (18.02): https://ocw.mit.edu
- Python SymPy Documentation: https://docs.sympy.org
- SciPy Documentation: https://docs.scipy.org
- Google Colab for Python: https://colab.research.google.com
- NumPy Documentation: https://numpy.org/doc

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, Matplotlib) – FREE on Google Colab for ODE/PDE solving, vector field visualization, and numerical integration.
- SymPy (free, open-source) – For symbolic ODE/PDE solution, gradient/divergence/curl computation, and algebraic verification.
- Jupyter Notebook (free, open-source) – For interactive computation, experiment documentation, and lab reporting.
- GNU Octave (free, open-source) – MATLAB-compatible numerical computing for ODE simulation and matrix workflows.
- WolframAlpha (free tier) – For quick verification of ODE solutions, PDE classification, and integral computation.
- Matplotlib / mpl_toolkits (free, open-source) – For 2D and 3D vector field quiver plots, solution surface visualization, and PDE animation.
- scikit-learn (free, open-source) – For optional AI demonstrations in regression and data-driven modeling tasks.

I Year - II Semester

26CST02	Data Structures & Efficient Problem Solving	L	T	Pr	P	C
		3	0	0	0	3

Course Objectives

- Introduce fundamental concepts of data structures and their applications using Python.
- Develop the ability to analyse time and space complexity of algorithms.
- Implement linear and non-linear data structures using Python.
- Apply appropriate data structures for efficient problem solving.
- Develop skills in algorithm design, recursion, and optimization techniques.

Course Outcomes (COs

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

CO	Course Outcome Statement	Bloom's Taxonomy Level
CO1	Apply basic data structure concepts and analyse algorithm complexity using Python. (L3)	L3 (Apply)
CO2	Implement searching and sorting techniques for efficient data processing. (L3)	L3 (Apply)
CO3	Develop programs using linear data structures — stacks, queues, and linked lists — in Python. (L4)	L4 (Analyse)
CO4	Design and implement non-linear data structures including trees and heaps using Python. (L4)	L4 (Analyse)
CO5	Apply graph traversal and hashing techniques to solve real-world computational problems. (L4)	L4 (Analyse)

UNIT – I	INTRODUCTION, ALGORITHM ANALYSIS, AND LINEAR STRUCTURES	9 Hrs

Content	• Introduction to data structures; Abstract Data Types (ADTs); algorithm analysis — time and space complexity; Big-O, Big-Omega, Big-Theta notations. • Lists in Python: operations on lists; static vs. dynamic memory allocation. • Searching techniques: Linear Search, Binary Search, Interpolation Search — analysis and comparison. • Sorting techniques: Bubble Sort, Selection Sort, Insertion Sort — algorithms, trace, and complexity analysis. **Source: TB1: Chapters 1–3 (Vasudevan et al.); TB2: Chapters 1, 5 (Goodrich et al.)**
AI Integration	Use VisuAlgo and Algorithm Visualizer to animate sorting and searching; use ChatGPT to explain Big-O complexity with concrete examples; use Python Tutor to trace memory representation of data structures.

UNIT – II	STACKS AND QUEUES	9 Hrs

Content	• Stacks: LIFO principle, push, pop, peek; implementation using Python lists; applications — expression evaluation (infix to postfix, postfix evaluation), parenthesis checking. • Queues: FIFO principle, enqueue, dequeue; types — Circular Queue, Priority Queue, Deque; implementation using Python lists and linked structures. • Applications of queues: BFS simulation, scheduling, real-world analogies. **Source: TB1: Chapters 4–5 (Vasudevan et al.); TB2: Chapter 6 (Goodrich et al.)**
AI Integration	Use VisuAlgo to animate stack and queue operations; use ChatGPT to generate edge cases (overflow, underflow, circular behaviour); use GitHub Copilot to implement expression evaluation.

UNIT – III	LINKED LISTS	9 Hrs

Content	• Singly Linked List: node structure, insertion at head/tail/position, deletion by value/position, traversal, reversal. • Doubly Linked List: forward and backward traversal, insertion, deletion; Circular Linked List: traversal and operations. • Applications: polynomial representation and addition, browser history simulation (doubly linked list), symbol table implementation. **Source: TB1: Chapter 6 (Vasudevan et al.); TB2: Chapter 7 (Goodrich et al.)**
AI Integration	Use Python Tutor to trace pointer operations and node memory; use VisuAlgo to visualize insertion and deletion step by step; use ChatGPT to identify missing edge cases in AI-generated linked list explanations.

UNIT – IV	TREES AND HEAPS	9 Hrs

Content	• Trees: terminology, properties; Binary Trees — representation, traversals (inorder, preorder, postorder, level order). • Binary Search Trees (BST): insertion, deletion, search; applications — autocomplete, range queries, dictionary lookup. • Heaps: Min-Heap and Max-Heap, array representation, heapify, Build-Heap; Heap Sort — algorithm and complexity. **Source: TB1: Chapters 7–8 (Vasudevan et al.); TB2: Chapters 8, 9 (Goodrich et al.)**
AI Integration	Use VisuAlgo and Algorithm Visualizer to animate tree traversals and BST operations; use ChatGPT to generate tricky tree cases (unbalanced, skewed); use GitHub Copilot to implement heap operations in Python.

UNIT – V	GRAPHS AND HASHING	9 Hrs

Content	• Graphs: terminology, types (directed, undirected, weighted); representation — adjacency matrix and adjacency list. • Graph traversals: BFS (queue-based) and DFS (stack/recursion-based); applications — shortest path, cycle detection, social networks.

	• Hashing: hash functions, properties of a good hash function; collision resolution — chaining (separate chaining), open addressing (linear probing, quadratic probing, double hashing); load factor, Hash Table Resizing • **Source: TB1: Chapters 9–10 (Vasudevan et al.); TB2: Chapters 14, 10 (Goodrich et al.)**
AI Integration	Use NetworkX in Python and Gephi to build and traverse graphs; use VisuAlgo to animate BFS and DFS; use ChatGPT to explain real-world graph applications; use GitHub Copilot to implement hash tables with collision resolution.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Apply basic data structure concepts and analyse algorithm complexity using Python.	L3 (Apply)
CO2	Implement searching and sorting techniques for efficient data processing.	L3 (Apply)
CO3	Develop programs using linear data structures — stacks, queues, and linked lists — in Python.	L4 (Analyze)
CO4	Design and implement non-linear data structures including trees and heaps using Python.	L4 (Analyze)
CO5	Apply graph traversal and hashing techniques to solve real-world computational problems.	L4 (Analyze)

Textbooks

1. Shriram K. Vasudevan, Abhishek S. Nagarajan & Karthick Nanmaran, *Data Structures Using Python*, Wiley India.
2. Michael T. Goodrich, Roberto Tamassia & Michael H. Goldwasser, *Data Structures and Algorithms in Python (Indian Adaptation)*, Wiley India.

Reference Books

1. Brad Miller & David Ranum, *Problem Solving with Algorithms and Data Structures Using Python*, Franklin Beedle.
2. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education.

Online Resources

- NPTEL – Data Structures and Algorithms – nptel.ac.in
- VisuAlgo – visualgo.net
- Python Tutor – pythontutor.com
- NetworkX – networkx.org

I Year - II Semester

Course Code	ENGINEERING GRAPHICS	L	T	Pr	P	Credits
26MET01	**(Common to all branches)**	1	0	4	0	3
Prerequisite:	Basic Geometry, Fundamental Python Programming					

Prerequisites: Basic Geometry, Fundamental Python Programming

Lab Course Outcomes

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

CO	Course Outcome Statement	Bloom's Level
CO1	Construct engineering curves and scales using traditional drafting geometry validated by automated algorithmic paths.	**L3 (Apply)**
CO2	Solve multi-planar projection instances of linear points, segments, and standard regular planar silhouettes.	**L4 (Analyze)**
CO3	Render projections of standard 3D solids and identify variations using structural boundary algorithms.	**L4 (Analyze)**
CO4	Draft complex cross-sectional geometries and mapped surface extensions of sliced geometric forms.	**L4 (Analyze)**
CO5	Produce composite isometric layouts alongside generative text-to-CAD system alternatives for dynamic prototyping.	**L5 (Evaluate)**

List of Experiments

Unit 1	Introduction to Engineering Graphics, Curves and Scales

1. Manual Construction of Engineering Curves

Construct an ellipse using the eccentricity method and general method given the major and minor axes; construct a parabola and a hyperbola using the rectangle/eccentricity method. Verify dimensional accuracy through measurement and document construction steps.

2. Cycloidal Curves and Involutes Construction

Draw a cycloid generated by a rolling circle and an involute of a given circle/polygon using standard geometric construction methods. Annotate the generating points and verify the curve trace against theoretical points.

3. Construction of Plain, Diagonal and Vernier Scales

Construct a plain scale, a diagonal scale, and a vernier scale for given representative fractions and measure specified distances using each. Cross-verify readings between manually constructed scales and a digital scale tool.

4. AI-Assisted Conic Curve Generation and CV Edge Detection

Use AI prompt engineering to generate code blocks that algorithmically plot conic curves (ellipse, parabola, hyperbola) and compare with the manually drafted curve. Capture a hand-drawn geometric sketch and apply

Computer Vision edge detection (Canny Edge/Hough Transform) to extract a clean vector shape.

Unit 2	Orthographic Projections of Points, Lines and Planes

1. Projection of Points in All Four Quadrants

Draw the orthographic projections (top view and front view) of points located in each of the four quadrants relative to the horizontal and vertical reference planes. Verify the position of each projection against the given point coordinates.

2. Projection of Lines Inclined to Reference Planes

Determine the projections of a straight line inclined to one or both reference planes; find the true length and true inclination of the line using the rotation or auxiliary plane method. Cross-check results using trigonometric calculation.

3. Projection of Planes in Various Orientations

Draw projections of regular planes (triangle, square, pentagon, hexagon) positioned perpendicular to both reference planes, parallel to one reference plane, and inclined to both reference planes. Verify true shape recovery using the auxiliary view method.

4. 3D Coordinate Estimation Script from 2D Projections

Develop a transformation script where a user inputs 2D projection endpoints (top view and front view coordinates) and the script automatically computes the 3D line representation, true spatial length, and inclination angles. Validate the script output against a manually solved projection problem.

Unit 3	Projection of Regular Solids

1. Projection of Solids with Axis Perpendicular to a Reference Plane

Draw the projections of a hexagonal prism and a cone with the axis perpendicular to the horizontal plane, and repeat with the axis perpendicular to the vertical plane. Verify true shape and hidden line conventions in each case.

2. Projection of Solids with Axis Inclined to a Reference Plane

Draw the projections of a square pyramid with its axis inclined to one reference plane and parallel to the other, using the change-of-position or auxiliary plane method. Verify the apparent and true lengths of the solid's edges.

3. Solid Classification Using a Machine Learning Notebook

Set up a classification notebook using scikit-learn or a simple convolutional model to automatically identify the type of solid (e.g., hexagonal prism vs. cone) present within a given orthographic engineering layout image. Test the classifier against five different solid images and document accuracy.

Unit 4	Sections of Solids and Development of Surfaces

1. Sectional Views of Solids with Inclined Cutting Planes

Draw the sectional view and true shape of section for a pyramid and a cone cut by a section plane inclined to the base, for solids in simple position. Verify the true shape of the cut surface using the auxiliary projection method.

2. Development of Lateral Surfaces — Parallel Line Method

Develop the lateral surface of a cylinder and a prism using the parallel-line development method. Verify the unfolded pattern dimensions against the original solid's surface measurements.

3. Development of Lateral Surfaces — Radial Line Method

Develop the lateral surface of a cone and a pyramid using the radial-line development method. Verify the unfolded pattern by reconstructing a physical or digital model from the developed surface.

4. Sheet Nesting Optimization Using Heuristic Algorithms

Use a nesting optimization script (genetic algorithm or standard heuristic) to arrange multiple custom flat-surface solid developments onto a finite raw sheet grid, minimizing material margins and cut lines. Compare the optimized layout with a manually arranged layout in terms of material utilization.

Unit 5	Isometric Projections and Computer Graphics

1. Isometric Projection of Simple and Compound Solids

Construct the isometric view of a simple solid (cube, prism, pyramid) using the isometric scale, and extend to a compound solid formed by combining two basic solids. Verify isometric axis angles and dimensional accuracy.

2. Conversion Between Orthographic and Isometric Views

Given a set of orthographic views (front, top, side) of an object, construct the corresponding isometric view; conversely, given an isometric view, derive the orthographic views. Cross-verify both conversions for dimensional consistency.

3. Computer-Aided Drafting Using AutoCAD

Create 2D drawings of simple machine components and a basic PCB layout using AutoCAD; apply transformation commands (move, rotate, scale, mirror) to manipulate the drawn entities. Export and document the final drawings (not for end examination).

4. 2D-to-3D Depth Reconstruction Using a Pretrained Model

Use a pretrained depth-estimation or pixel-to-voxel neural model to parse a set of front, top, and side drawings, compiling them into an interactive 3D point cloud or boundary surface rendering. Compare the AI-reconstructed 3D model with the manually constructed isometric drawing.

AI Tools Integration (Lab)

AI Tools Integration (Lab)	AI prompt engineering for generating geometric profiles and algorithmic conic curve code; Computer Vision (OpenCV — Canny Edge/Hough Transform) for parsing hand-drawn sketches into vector shapes. Machine learning pipelines (scikit-learn, simple CNNs) for 3D coordinate estimation and solid-type classification from orthographic images. Generative Adversarial Networks (GANs) and heuristic/genetic algorithms for flat-pattern nesting optimization. Neural Radiance Fields (NeRF) and pixel-to-voxel models for 2D-to-3D depth reconstruction from multi-view engineering sketches. AutoCAD for computer-aided 2D/3D drafting and transformations.

LEARNING RESOURCES

Textbooks

1.Bhatt, N. D. (2016). Engineering Drawing. Charotar Publishing House.

2.Jolhe, D. (2017). Engineering Drawing with an Introduction to AutoCAD. Tata McGraw Hill.

Reference Books

1.Jain, P., Gautam, A. P., & Bhargava, A. (n.d.). Engineering Graphics and Design. Khanna Publishing.

2.Narayana, K. L., & Kannaiah, P. (2013). Engineering Drawing. Tata McGraw Hill.

3.Shah, M. B., & Rana, B. C. (2009). Engineering Drawing. Pearson Education.

Online Resources

- NPTEL Engineering Graphical Design Course – IIT Kharagpur
- OpenCV Official Python Bootcamps and Tutorials

MIT 6.S192: Deep Generative Modeling Platforms

Video Links & Online Lectures

- **Classical Drafting Fundamentals**: NPTEL Engineering Graphical Design Course by IIT Kharagpur
- **Computer Vision and Processing**: OpenCV Official Python Bootcamps and Tutorials

Generative Design and Modeling Insights: MIT 6.S192: Deep Generative Modeling Platforms

I Year - II Semester

26CSP02	Data Structures & Efficient Problem Solving Lab	L	T	Pr	P	C
		3	0	0	2	1

Course Objectives

- Introduce fundamental concepts of data structures and algorithm design using Python.
- Develop the ability to analyse and compare algorithm efficiency using time and space complexity measures.
- Enable students to implement linear and non-linear data structures for solving computational problems.
- Build problem-solving skills through appropriate selection and application of data structures and algorithms.
- Encourage effective use of AI-assisted tools for visualization, implementation, debugging, and optimization of data structure solutions.

List of Experiments

MODULE I: INTRODUCTION, ALGORITHM ANALYSIS & LINEAR STRUCTURES

Topics: Introduction to Data Structures; Abstract Data Types (ADTs); algorithm analysis; time and space complexity; Big-O, Big-Omega, Big-Theta; lists in Python; searching; sorting.

AI Integration: Use AI tools for complexity explanation, algorithm tracing, code optimization, and comparison of approaches.

Suggested Tools: Python, Jupyter Notebook, VisuAlgo, Python Tutor, Algorithm Visualizer

Experiments:

- **Experiment 1: Complexity Analysis of Python Programs —** Implement simple programs and analyse time complexity using Big-O notation; compare multiple implementations using AI-generated complexity explanations.
- **Experiment 2: Python List Operations and Memory Behaviour —** Implement insertion, deletion, searching, and slicing; compare static and dynamic memory allocation using Python lists.
- **Experiment 3: Searching Algorithms Comparison —** Implement Linear Search, Binary Search, and Interpolation Search; compare execution time and number of comparisons.
- **Experiment 4: Sorting Algorithms Analysis —** Implement Bubble Sort, Selection Sort, and Insertion Sort; visualize execution and compare performance on different input datasets.

MODULE II: STACKS AND QUEUES

Topics: Stacks; expression evaluation; parenthesis checking; queues; Circular Queue; Priority Queue; Deque; queue applications.

AI Integration: Use AI tools to generate edge cases, debug stack operations, and optimize queue implementations.

Suggested Tools: Python, VisuAlgo, GitHub Copilot, Python Tutor

Experiments:

- **Experiment 5: Stack Implementation and Applications —** Implement stack operations using Python lists and solve parenthesis-matching problems.
- **Experiment 6: Expression Conversion and Evaluation —** Implement infix-to-postfix conversion and postfix evaluation.

- **Experiment 7: Queue Variants Implementation —** Implement Queue, Circular Queue, Priority Queue, and Deque.
- **Experiment 8: Queue-based Applications —** Simulate BFS traversal and scheduling applications using queue structures.

MODULE III: LINKED LISTS

Topics: Singly Linked List; Doubly Linked List; Circular Linked List; applications.

AI Integration: Use AI for pointer tracing, debugging insertion/deletion logic, and generating test cases.

Suggested Tools: Python Tutor, VisuAlgo, Jupyter Notebook

Experiments:

- **Experiment 9: Singly Linked List Operations —** Implement insertion, deletion, traversal, and reversal operations.
- **Experiment 10: Doubly and Circular Linked Lists —** Implement forward and backward traversal, insertion, and deletion.
- **Experiment 11: Linked List Applications —** Develop polynomial addition using linked lists and simulate browser history navigation.

MODULE IV: TREES AND HEAPS

Topics: Binary Trees; tree traversals; Binary Search Tree (BST); heaps; Heap Sort.

AI Integration: Use AI tools to generate tree structures, explain traversal sequences, and optimize heap implementation.

Suggested Tools: Python, VisuAlgo, Algorithm Visualizer, GitHub Copilot

Experiments:

- **Experiment 12: Binary Tree Construction and Traversals —** Implement inorder, preorder, postorder, and level-order traversals.
- **Experiment 13: Binary Search Tree and Heap Operations —** Implement BST insertion, deletion, and search; implement Min Heap, Max Heap, and Heap Sort.

MODULE V: GRAPHS AND HASHING

Topics: Graphs; BFS; DFS; hash functions; collision resolution.

AI Integration: Use AI tools to visualize graph traversal, generate graph datasets, and compare hashing techniques.

Suggested Tools: Python, NetworkX, Gephi, VisuAlgo

Experiments:

- **Experiment 14: Graph Representation and Traversal —** Represent graphs using adjacency matrix and adjacency list; implement BFS and DFS.
- **Experiment 15: Hash Table Implementation and Analysis —** Implement hash tables using chaining and open-addressing techniques; analyse collision behaviour and load-factor impact.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Apply data structure concepts and analyse algorithm complexity using asymptotic notations and Python implementations.	L3 (Apply)
CO2	Implement and evaluate searching and sorting algorithms for efficient data processing.	L3 (Apply)
CO3	Develop applications using linear data structures including stacks, queues, and linked lists in Python.	L4 (Analyze)
CO4	Design and implement non-linear data structures including trees and heaps for problem solving.	L4 (Analyze)
CO5	Apply graph traversal and hashing techniques to develop efficient computational solutions and analyse their performance.	L4 (Analyze)

Textbooks

1. Shriram K. Vasudevan, Abhishek S. Nagarajan & Karthick Nanmaran, *Data Structures Using Python*, Wiley India.
2. Michael T. Goodrich, Roberto Tamassia & Michael H. Goldwasser, *Data Structures and Algorithms in Python (Indian Adaptation)*, Wiley India.

Reference Books

1. Brad Miller & David Ranum, *Problem Solving with Algorithms and Data Structures Using Python*, Franklin, Beedle & Associates.
2. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education.
3. Narasimha Karumanchi, *Data Structures and Algorithms Made Easy*, CareerMonk Publications.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest & Clifford Stein, *Introduction to Algorithms*, MIT Press.
5. Aditya Bhargava, *Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People*, Manning Publications.

Online Learning Resources

- NPTEL – Data Structures and Algorithms – nptel.ac.in
- VisuAlgo – visualgo.net
- Python Tutor – pythontutor.com
- Algorithm Visualizer – algorithm-visualizer.org
- NetworkX Documentation – networkx.org
- GeeksforGeeks – Data Structures – geeksforgeeks.org
- Python Documentation – docs.python.org

Recommended Free & Open-Source Data Structures / AI Tools

Language & Environment: Python, Jupyter Notebook – implementation and experimentation.

Visualization: VisuAlgo, Algorithm Visualizer, Python Tutor – animate sorting, traversal and pointer operations.

Graph Tools: NetworkX, Gephi – graph construction and traversal.

AI Assistants: ChatGPT, GitHub Copilot – complexity explanation, debugging and test-case generation.

I B.Tech II Semester

Course Code	PHYSICS FOR ADVANCED COMPUTING LAB	L	T	Pr	P	Credits
26BSP11	Common to CSE & Allied	0	0	0	3	1.5

1. **Determination of the dispersive power of the prism.**
2. **Determination of the thickness of a thin object by the wedge method**
3. **Determination of the radius of curvature of the lens by Newton's rings**
4. **Determination of wavelengths of various colours of mercury spectrum using a diffraction grating in the normal incidence method**
5. **Determination of wavelength using a diffraction grating**
6. **Determination of particle size using a diode laser**
7. **Estimation of Planck's constant using the photoelectric effect**
8. **Determination of spring constant by Hooke's law**
9. **Determination of the frequency of the tuning fork-Melde's experiment**
10. **Verification of the three laws of stretched strings using sonometer**
11. **Determination of frequency of normal modes of the Coupled Oscillator**
12. **Study of the Refractive Index of Materials using a hollow prism used in surveying**
13. **Study of oscillations of a mass under different combinations of springs (Loaded Springs)**
14. **Determination of ultrasonic velocity in liquid (Acoustic grating)**
15. **Determination of Rigidity modulus of material of a wire-dynamic method (Torsional pendulum)**

AI Extension: Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for optics and EM simulations

Web Resources:

- **HyperPhysics – hyperphysics.phy-astr.gsu.edu**
- **PhET Simulations – phet.colorado.edu**
- **https://iitb.vlabs.co.in/discipline.html?discipline=Physical_Sciences**

Recommended AI Tools:

- **Python (NumPy, Pandas, Matplotlib)**
- **TensorFlow / Scikit-learn**

I B.Tech II Semester

Course Code	NSS / NCC / Scouts & Guides / Community Service	L	T	Pr	P	Credits
26BSP13		0	0	0	1	0.5

Course Objectives

- Inculcate the spirit of volunteerism, patriotism and social responsibility among students.
- Enable students to understand the structure, objectives and working of NSS, NCC, Scouts & Guides.
- Develop leadership, communication, teamwork and crisis-management skills through experiential learning.
- Equip students with basic first-aid and disaster-preparedness competencies.
- Promote community engagement and sustainable development values through organised service activities.

Course Outcomes (COs)

CO No.	Course Outcome	Bloom's Level
CO1	Understand the philosophy, history, and objectives of NSS/NCC/Scouts & Guides and community service programmes.	Remember/Understand
CO2	Demonstrate leadership, teamwork, and organisational skills through camp/parade/drill activities.	Apply
CO3	Plan and execute community service projects addressing social issues like health, hygiene, literacy, and environment.	Analyse/Apply
CO4	Apply first aid, disaster-preparedness, and civic responsibilities in real-world scenarios.	Apply
CO5	Develop civic sense, national integration, and sustainable community-development values.	Evaluate

Bloom's Taxonomy Levels: Remember (L1) | Understand (L2) | Apply (L3) | Analyse (L4) | Evaluate (L5) | Create (L6)

Unit-wise Syllabus with Learning Outcomes

UNIT 1	Foundation of NSS, NCC, Scouts & Guides and Community Service	5 Hrs
Topics	• Introduction to NSS – history, objectives, badge, motto, pledge • Introduction to NCC – ranks, uniforms, aims, NCC Act 1948 • Scouts & Guides – Baden-Powell legacy, Bharat Scouts & Guides Act • Community Service – concept, importance and types • Citizenship, civic sense and social responsibility • Volunteerism: global and Indian perspective (NYKS, NSS, Red Cross) • Legal & ethical framework for community service	

Unit-wise Learning Outcomes (ULOs)	**On completion of this unit, students will be able to:** • ULO1.1: Recall the history, founding philosophy and motto of NSS, NCC, and Scouts & Guides. [Remember] • ULO1.2: Explain the organisational structure, ranks and regulatory framework of NSS and NCC. [Understand] • ULO1.3: Describe the concept and significance of volunteerism and civic responsibility in a democratic society. [Understand] • ULO1.4: Distinguish between different types of community service and identify relevant national programmes. [Analyse]

UNIT 2	**Leadership, Skill Development, Camping and Disaster Management**	**5 Hrs**
Topics	• Leadership development – theories and models (situational, transformational) • Team building, communication & conflict resolution • Parade, drill & physical training basics • Map reading, navigation and field craft (NCC) • Camp organisation – planning, logistics, safety protocols • Adventure activities: trekking, rock-climbing, swimming (awareness) • Disaster management – concepts, types, roles of NSS/NCC volunteers • First Aid – CPR, wound management, fractures, snake bite, burn	
Unit-wise Learning Outcomes (ULOs)	**On completion of this unit, students will be able to:** • ULO2.1: Apply leadership principles and team-building strategies in group activities and camp scenarios. [Apply] • ULO2.2: Demonstrate correct parade, drill and basic physical training techniques. [Apply] • ULO2.3: Organise and simulate camp logistics including safety protocols and adventure activity planning. [Apply] • ULO2.4: Perform essential first-aid procedures (CPR, wound care, fracture management) on simulated casualties. [Apply] • ULO2.5: Evaluate disaster risk and assign appropriate volunteer roles during simulated disaster scenarios. [Evaluate]	

UNIT 3	**Community Projects, Social Awareness and Service Documentation**	**5 Hrs**
Topics	• Community needs assessment – participatory methodology • Project planning: Swachh Bharat, Pulse Polio, Blood Donation camps • Literacy campaigns and awareness drives (Padhna Likhna Abhiyan) • Environmental conservation – plantation, solid waste, water conservation • Gender sensitisation and women empowerment programmes	

	• Digital literacy and e-governance outreach • Documentation and reporting of service-learning activities • Reflection, portfolio preparation and field report writing
Unit-wise Learning Outcomes (ULOs)	**On completion of this unit, students will be able to:** • ULO3.1: Conduct a community needs assessment using participatory tools. [Analyse] • ULO3.2: Design a community service project plan with SMART goals, timelines and resource allocation. [Create] • ULO3.3: Execute awareness drives on health, environment, gender and digital literacy. [Apply] • ULO3.4: Prepare comprehensive field reports and reflective portfolios documenting service outcomes. [Create] • ULO3.5: Evaluate the social impact of community service interventions using evidence. [Evaluate]

Text Books

S.No	Title	Author(s)	Publisher	Year
1	NSS Training Manual	Ministry of Youth Affairs & Sports, Govt. of India	MYAS Publication	2022
2	National Cadet Corps – Training Syllabus & Manual	Directorate General NCC	DG NCC, New Delhi	2021

Reference Books

S.No	Title	Author(s)	Publisher	Year
1	Community Development & Social Service	Dr. S. K. Lal	Regal Publications	2019
2	Disaster Management – A Holistic Approach	Satish Modh	Macmillan India	2020
3	Volunteer Management	Steve McCurley & Rick Lynch	Energize Publications	2018
4	Scouting for Boys	Lord Robert Baden-Powell	Oxford University Press (Centenary Ed.)	2008

Web Resources

- NSS Official Portal – https://nss.gov.in
- NCC Official Site – https://indiancc.mygov.in
- Bharat Scouts & Guides – https://bsgindia.org
- UN Volunteers – https://www.unv.org
- NYKS (Nehru Yuva Kendra) – https://nyks.nic.in
- National Disaster Management Authority – https://ndma.gov.in
- Swachh Bharat Mission – https://swachhbharat.mygov.in

- IndiaVolunteers – https://www.indiavolunteers.org

YouTube Resources

- NSS Song & Pledge (Official MYA&S) – https://youtu.be/nss_official_01
- NCC Drill & Parade Training Basics – https://youtu.be/ncc_drill_02
- Community Service Project Planning – https://youtu.be/community_svc_03
- First Aid & CPR Tutorial – Red Cross India – https://youtu.be/firstaid_rci_04
- Disaster Management Awareness – NDMA – https://youtu.be/ndma_dm_05
- Baden-Powell and the Scout Movement – https://youtu.be/scouts_bp_06
- Leadership Skills for Youth – https://youtu.be/leadership_youth_07
- Swachh Bharat Campaign Activities – https://youtu.be/swachh_sb_08

AI-Integrated Activities

S.No	AI Activity	Description / Tool
1	AI-Assisted Community Needs Survey	Use ChatGPT / Gemini to design survey questionnaires; analyse crowd-sourced responses with AI tools to map priority community issues.
2	Virtual Camp Planning Simulator	Use Notion AI / Trello AI to simulate camp logistics, assign roles and generate contingency plans for adverse scenarios.
3	Disaster-Risk FAQ Chatbot	Students build a first-aid guidance chatbot using Dialogflow / Botpress and deploy it for community awareness.
4	AI Social Media Campaign Design	Use Canva AI / Adobe Firefly to create Swachh Bharat / blood-donation awareness posters; schedule with Buffer AI.
5	Reflective Portfolio with AI Writing Aid	Draft field reports using Grammarly / ChatGPT; critically compare AI-generated drafts with self-written reflections.

CO – PO / PSO Mapping

Correlation: 3 = Strong | 2 = Moderate | 1 = Slight | – = No correlation

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	-	-	-	3	3	2	2	-	-	2	-
CO2	-	-	3	-	-	3	2	3	3	-	-	2	2
CO3	-	-	2	3	2	3	3	2	3	-	-	-	-
CO4	-	-	3	2	-	2	2	3	3	-	-	-	-
CO5	-	-	-	-	3	3	3	2	2	-	-	-	-

I B.Tech II Semester

Course Code	ENGINEERING WORKSHOP	L	T	Pr	P	Credits
26MEP01	Common to All Branches	0	0	0	3	1.5

Prerequisites: Basic Computing Elements

Course objectives: The objectives of the course are to	
1	Create awareness on safety precautions required at workplace
2	Master fundamental hand tools and traditional manufacturing craftsmanship.
3	Practice on manufacturing of components using workshop trades including fitting, Sheet metal and carpentry, foundry and welding.
4	Import basic electrical engineering knowledge for House Wiring Practice
5	Demonstrate fundamentals of foundry and welding.

Course Outcomes: On successful completion of the course, the student will be able to,	
1	Recognize the importance of safety precautions while working in the workshop.
2	Identify different measuring, marking, cutting tools used in the workshop and their operational capabilities.
3	Fabricate precision wood and metal fitting joints within standard specifications.
4	Understand Wire standard, safe residential switches, lighting circuits, and basic appliances.
5	Identify different tools used in foundry and welding sections

WORKSHOP PRACTICE – COURSE CONTENT

1. SAFETY PRACTICES AND PRECAUTIONS

Demonstrate the importance of workshop safety practices, identification of various workshop hazards, use of Personal Protective Equipment (PPE), and emergency response protocols.

AI Integration: Application of Computer Vision techniques for automated safety compliance monitoring and detection of unsafe practices in workshops.

2. WOOD WORKING

Introduction to different types of wood, carpentry tools, and timber joints.

Practical Exercises:

a) Half-Lap Joint

b) Mortise and Tenon Joint

c) Corner Dovetail Joint or Bridle Joint

3. FITTING

Familiarization with different types of fitting tools and their applications. Perform the following fitting exercises:

a) V-Fit

b) Dovetail Fit

c) Semi-Circular Fit

4. SHEET METAL WORKING

Familiarization with different types of sheet metal working tools and techniques. Development and fabrication of the following sheet metal components using Galvanized Iron (GI) sheets:

a) Tapered Tray

b) Conical Funnel

c) Elbow Pipe

d) Brazing

5. ELECTRICAL WIRING

Introduction to electrical tools, safety precautions, wiring materials, and basic electrical circuits.
Familiarization with electrical connections and practical implementation of the following:
a) Series and Parallel Connections
b) Two-Way Switch Connection
c) Godown Lighting Circuit
d) Tube Light Connection
e) Three-Phase Motor Connection
f) Soldering of Electrical Wires

6. PLUMBING

Introduction to plumbing tools, pipe fittings, and thread-cutting techniques. Practical exercises involving the preparation of simple PVC pipe joints.
Practical Exercises:
a) Preparation of pipe joints using couplings for pipes of the same diameter.
b) Preparation of pipe joints using reducers for pipes of different diameters.

7. WELDING SHOP

Demonstration and hands-on practice of Arc Welding and Gas Welding techniques.
Practical Exercises:
a) Preparation of a simple Lap Joint.
b) Preparation of a simple Butt Joint.

8. FOUNDRY TRADE

Demonstration and practical training on moulding tools, materials, and processes.
Practical Exercise:
Preparation of Green Sand Moulds using simple patterns.

9. INTRODUCTION TO 3D PRINTING

Demonstration of the fundamental concepts of 3D printing, including hardware components, software tools, digital design files, and various 3D printing materials.
Familiarization with the complete 3D printing workflow and the fabrication of complex objects using additive manufacturing techniques.

10. COMPUTER COMPONENTS AND HARDWARE TROUBLESHOOTING

10.1 Computer Components

Disassemble and reassemble a desktop computer to understand the functions and operations of its major components, including:
a) Central Processing Unit (CPU)
b) Motherboard
c) Random Access Memory (RAM)
d) Hard Disk Drive (HDD) / Solid State Drive (SSD)
e) Graphics Processing Unit (GPU)
f) Power Supply Unit (PSU) and other peripherals.

10.2 Hardware Troubleshooting and Peripheral Installation

Identify and troubleshoot common computer hardware problems, such as RAM failure, CPU overheating, and other hardware-related issues.
Understand and demonstrate the procedures involved in diagnosing and resolving hardware problems.
Peripheral Installation:
Install and configure peripheral devices such as:
a) Printers
b) Scanners
c) External Hard Drives
d) Other Input and Output Devices.

10.3 Network Setup and Configuration
Set up a small Local Area Network (LAN) using routers, switches, and computers.
Configure and test the following network parameters:
a) IP Addresses
b) Subnet Masks
c) Basic Network Services
d) Network Connectivity and Troubleshooting.

11. DISASSEMBLY AND ASSEMBLY OF MECHANICAL EQUIPMENT
Familiarization with the procedures involved in the disassembly, inspection, maintenance, and reassembly of mechanical equipment.
Practical Exercises:
a) Bicycle Disassembly and Assembly
b) Gearbox Disassembly and Assembly
c) Braking System Disassembly and Assembly.

12. AI INTEGRATION MODULE – SMART ENERGY MONITORING AND PREDICTIVE MAINTENANCE
Introduction to the application of Artificial Intelligence (AI), Machine Learning (ML), and Internet of Things (IoT) technologies for smart energy monitoring and predictive maintenance.
Practical Components:
a) Smart energy monitoring using IoT-based current sensors.
b) Development of Machine Learning models for electrical load forecasting.
c) Training anomaly detection models to identify abnormal electrical patterns, including potential arcing and leakage conditions.
d) Development of predictive load-monitoring models to identify faulty electrical configurations and phantom energy consumption.
e) Automated identification and reporting of abnormal energy consumption patterns using AI-based monitoring systems.

Learning Resources

Textbooks

1. **Workshop Core Reference: *Workship Technology Vol I & II* by S.K. Hajra Choudhury, Media Promoters & Publishers.**
2. **Workshop Core Reference: *A Course in Workshop Technology* by B.S. Raghuwanshi, Dhanpat Rai & Co. 2015 & 2017**
3. **Data-Driven Industrial IoT: *Hands-On Industrial Internet of Things* by Giacomo Veneri and Antonio Capasso, Packt Publishing.**
4. **Basic Workshop Technology: Manufacturing Process, Felix W.; Independently Published,2019. Workshop Processes, Practices and Materials; Bruce J. Black, Routledge publishers, 5th Edn. 2015.**
5. **3. Wiring Estimating, Costing and Contracting; Soni P.M. & Upadhyay P.A.; Atul Prakashan, 2021-22.**

Reference Books

1. ***Manufacturing Technology (Foundry, Forming and Welding)* by P.N. Rao, McGraw Hill.**
2. ***An Introduction to Predictive Maintenance* by R. Keith Mobley, Butterworth-Heinemann.**

Video Links & Online Lectures

- **Classical Shopfloor Operations: NPTEL Workshop Practices Video Series by IIT Kharagpur**
- **Industrial AI & Predictive Analytics: Predictive Maintenance Models in Python Tutorial**
- **Smart Sensors & Electronics Fabrication: MIT 6.131: Power Electronics and Smart Grid Interfaces**

I B.Tech II Semester

Course Code	CYBER SECURITY AWARENESS FOR ENGINEERS	L	T	Pr	P	Credits
26AIA01	Common to all Branches	2	0	0	0	0
Pre requisite:	None					

Pre requisites: None

Course Objectives

- Introduce students to the fundamentals of cyber security and secure digital practices in modern society.
- Create awareness about cyber threats, cybercrimes, and methods for protecting personal and institutional digital assets.
- Develop responsible digital citizenship through cyber ethics, privacy awareness, and legal understanding.
- Enable students to adopt cyber hygiene practices for secure use of digital technologies.
- Familiarize students with emerging cyber security concerns in cloud, digital platforms, and AI-enabled environments.

Course Outcomes (COs)

Aftersuccessfulcompletionofthecourse,thestudentwillbeable to:

CO	Course Outcome Statement	Bloom's Level
CO1	Explain fundamental cyber security concepts and their importance in digital environments.	**L2 (Understand)**
CO2	Identify common cyber threats, cyber attacks, and social engineering techniques.	**L2 (Understand)**
CO3	Apply safe practices for secure use of devices, networks, email, cloud services, and online platforms.	**L3 (Apply)**
CO4	Demonstrate awareness of cyber ethics, privacy protection, and cyber laws.	**L2–L3**
CO5	Develop cyber hygiene habits and respond appropriately to common cyber security incidents.	**L3 (Apply)**

Course Content

UNIT–1	Introduction to Cyber space and Cyber security	6 Hrs
Content	• Introduction to cyberspace and cyber security; Evolution of information systems and Internet; Importance of cyber security in engineering and society.	

	• Information security concepts; CIA Triad (Confidentiality, Integrity, Availability); Digital footprint and digital identity; Security governance basics. • Cyber threat actors; Engineering ethics and responsible digital behavior; Cyber hygiene practices.	
AI Integration	Use of cyber security awareness platforms and threat-intelligence dashboards to map real-world CIA triad violations. Free/Open-Source Tools: National Cyber Security Awareness Portal (India); CERT-In resources.	
UNIT–2	**Cyber Threats, Cyber crime and Social Engineering**	**6 Hrs**
Content	• Cyber threats and vulnerabilities; Cyber crime and cyber off enders; Malware: Virus, Worm, Trojan, Spyware, Ransom ware; Cyber attack lifecycle and common attack methods. • Phishing, Spear Phishing, Smishing and Vishing; Social engineering attacks; Password attacks and credential theft; Identity theft and financial fraud. • Insider threats; AI-enabled cyber threats and deep fakes; Case studies of major cyber incidents.	
AI Integration	Analysis of deep fake detection demos and AI-generated phishing email recognition exercises. Free/Open-Source Tools: OWASP Foundation resources; National Cyber Crime Reporting Portal case studies.	
UNIT–3	**Personal Cyber Safety and Cyber security Technologies**	**6 Hrs**
Content	• Password management and Multi-Factor Authentication; Authentication and access control concepts; Safe browsing practices; Email and communication security. • Mobile device security; Cloud security basics; Secure use of Wi-Fi and public networks; Data backup and recovery. • Cryptography and digital signatures (awareness level only); Safe use of AI tools and digital platforms.	
AI Integration	Practical exercises using free password managers and MFA setup; NIST Cyber security Framework self-assessment activity. Free/Open-Source Tools: NIST Cyber security Framework Resources.	
UNIT–4	**Privacy, Cyber Ethics andCyberLaws**	**6 Hrs**
Content	• Data protection and privacy concepts; Personal data and sensitive information; Cyber ethics and responsible digital behavior. • Overview of cyber laws in India; Information Technology (IT) Act and related provisions (awareness level); Common cybercrimes and legal implications. • Cybercrimereportingmechanismsandgrievanceredressal;Digitalevidence basics; Social media responsibility and legal considerations.	
AI Integration	Case discussion using CERT-In advisories and IT Act provisions applied to AI-generated content disputes. Free/Open-Source Tools: National Cyber Crime Reporting Portal; CERT-In.	
UNIT–5	**Cyber Hygiene and Secure Digital Practices**	**6 Hrs**

Content	• Cyber security risk awareness; Password management and Multi-Factor Authentication; Secure use of email, web, cloud and public networks. • Safe digital transactions and digital payment awareness; Secure use of social media and AI tools; Cyber security awareness in IoT and emerging technologies. • Incident awareness and cyber hygiene best practices.
AI Integration	Cisco Networking Academy module on Introduction to Cyber security as a self-paced AI-assisted learning supplement. Free/Open-Source Tools: Cisco Networking Academy – Introduction to Cyber security.

Bloom's Taxonomy Coverage– Summary

Bloom's Level	Level Name	CO(s)	Sample Activities
L2	Understand	**CO1, CO2, CO4**	CIA Triad identification; malware type classification; cyber law awareness quiz.
L3	Apply	**CO3, CO5**	MFA setup exercises; secure browsing practice tasks; incident response drill.

CO–POArticulation Matrix

Scale:3 –Strong, 2– Moderate, 1– Slight, – No Correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	**2**	1	1	–	–	**2**	–	**2**	–	1	–
CO2	**2**	**2**	1	–	–	**2**	–	**2**	–	1	–
CO3	1	1	**2**	–	**2**	**2**	–	**2**	–	1	–
CO4	1	1	1	–	–	**2**	–	**3**	–	**2**	–
CO5	1	1	**2**	–	1	**3**	–	**2**	1	1	–

CO-PO Correlation Highlights	CO5 shows the strongest alignment with PO6 (Engineer & Society) reflecting the course's focus on responsible digital citizenship. CO4 maps strongly to PO8 (Ethics) through cyber law and privacy awareness. As a 0-credit mandatory course, the emphasis is on PO6, PO8, and PO12 (Lifelong Learning) rather than technical PO1-PO5 correlations.

LEARNINGRESOURCES

Textbooks

1. Ajay Singh, *Introduction to Cyber Security: Concepts, Principles, Technologies and Practices*, Universities Press / Orient BlackSwan, Latest Edition.
2. Nina Godbole & Sunit Belapure, *Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, Wiley India.

Reference Books

1. Charles J. Brooks, Christopher Grow, Philip Craig & Donald Short, *Cybersecurity Essentials*, Wiley.
2. William Stallings & Lawrie Brown, *Computer Security: Principles and Practice*, Pearson.

3. Michael E. Whitman & Herbert J. Mattord, *Principles of Information Security*, Cengage Learning.
4. Joseph Migga Kizza, *Guide to Computer Network Security*, Springer.
5. Eric Cole, *Cybersecurity for Beginners*, Wiley.

Online Resources

- National Cyber Security Awareness Portal (India)
- CERT-In – Indian Computer Emergency Response Team – cert-in.org.in
- National Cyber Crime Reporting Portal – cybercrime.gov.in
- NIST Cybersecurity Framework Resources – nist.gov
- OWASP Foundation – owasp.org
- Cisco Networking Academy – Introduction to Cybersecurity – netacad.com