SRI SAI INSTITUTE OF TECHNOLOGY AND SCIENCE

ESTD.:2001

AUTONOMOUS

25 YEARS of Academic EXCELLENCE

(Approved by AICTE, New Delhi & Affiliated to JNTUA, Ananthapuramu)
Rayachoty – 516 270, Annamayya (Dist.) - (A.P.)

INSTITUTION'S INNOVATION COUNCIL

NAAC B+

Course Structure and Detailed Syllabi for I, II, III & IV Year I & II Semesters

B.Tech.

Electrical and Electronics Engineering

SSITS R26 Regulations

A Four - Year Regular Degree Program

(Applicable for the batches amitted from the Academic Year 2026 - 27) &

A Three-Year Regular Degree Program under the Lateral Entry Scheme (LES)
(Applicable for the batches admitted from the Academic Year 2027 - 28) academic year 2026-27nwards)

SRI SAI INSTITUTE OF TECHNOLOGY AND SCIENCE
(AUTONOMOUS)
(Approved by AICTE New Delhi & Affiliated to J N T University, Anantapuramu & Accredited by NAAC)
Masapeta, RAYACHOTY - 516 270, Annamayya District, A.P.
Estd. 2001

B. Tech (Regular-Full time)

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

ELECTRICAL & ELECTRONICS ENGINEERING

I & II YEARS 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)

I Year I Semester

S.No.	Course code	Title	L	T	Pr	P	Credits
1	26AHS00105T	Engineering Chemistry For Electrical Engineers	3	0	0	0	3
2	26AHS00102T	Linear Algebra & Calculus	3	0	2	0	4
3	26ACS05101T	Design Thinking	1	0	4	0	3
4	26ACS05102T	Computational Thinking and Problem Solving with Python	3	0	0	0	3
5	26ACS05103T	AI Tools and Applications	1	0	4	0	3
6	26AME03101P	Engineering Workshop	0	0	0	3	1.5
7	26AHS00105P	Engineering Chemistry for Electrical Engineers Lab	0	0	0	3	1.5
8	26ACS05102P	Computational Thinking and Problem Solving with Python Lab	0	0	0	3	1.5
9	26AHS00101L	NSS/NCC/Scouts & Guides/Community Service	0	0	0	1	0.5
Total			**11**	**0**	**10**	**10**	**21**

I Year II Semester

S.No.	Course code	Title	L	T	Pr	P	Credits
1	26AHS00201T	Communicative English	2	0	0	0	2
2	26AHS00205T	Physics for Electrical Systems	3	0	0	0	3
3	26AHS00203T	Differential Equations & Vector Calculus	3	0	2	0	4
4	26AEE02201T	**Professional core - I** • Electromagnetic Fields	3	0	0	0	3
5	26AME03201T	Engineering Graphics	1	0	4	0	3
6	26AEE02206P	**Department Specific Workshop** • Electrical Engineering Workshop Practices	0	0	0	2	1
7	26AHS00205P	Physics for Electrical Systems Lab	0	0	0	3	1.5
8	26AHS00201P	Communicative English Lab	0	0	0	2	1
9	26AHS00201L	Health and wellness, Yoga and Sports	0	0	0	1	0.5
10	26ACS05101M	Cybersecurity Awareness for Engineers	2	0	0	0	0
Total			**14**	**0**	**6**	**8**	**19**

II Year I Semester

S.No.	Course code	Title	L	T	Pr	P	Credits
1	26AHS00301T	Complex Variables and Probability Theory	3	0	0	0	3
2	26AHS00302T	Universal Human Values – Understanding Harmony and Ethical Human Conduct	3	0	0	0	3
3	26AMB02301T	Managerial Economics and Financial Analysis	2	0	0	0	2
4	26AEC04304T	**Engineering Course** • Analog Circuits	3	0	0	0	3
5	26AEE02301T	**Professional core - II** • AI Assisted Electrical Circuit Analysis	2	0	2	0	3
6	26AEE02302T	**Professional core - III** • DC Machines and Transformers with Intelligent Control	2	0	2	0	3
7	26AEC04304P	**Engineering Course Lab** • Analog Circuits Lab	0	0	0	2	1
8	26AEE02301P	**Professional core - II Lab** • AI Assisted Electrical Circuit Analysis Lab	0	0	0	3	1.5
9	26AEE02303P	**Professional core - III Lab** • Intelligent Control of DC Machines and Transformers Lab	0	0	0	3	1.5
10	26ACS05301S	**Skill Enhancement course - I** • C Programming and Data Structures	1	0	0	2	2
11	26AHS00301M	Environmental Science	2	0	0	0	0
		Total	**18**	**0**	**4**	**10**	**23**

II Year II Semester

S.No.	Course code	Title	L	T	Pr	P	Credits
1	26AHS0304T	Biology for Engineers	2	0	0	0	2
2	26AME3401T	Sustainability and Green Engineering	2	0	0	0	2
3	26AEE02401T	**Professional core - IV** • AI Assisted Control Systems	2	0	2	0	3
4	26AEE02403T	**Professional core - V** • AI-Assisted Control of Induction and Synchronous Machines	2	0	2	0	3
5	26AEE02404T	**Professional core - VI** • Intelligent Power Systems – I	2	0	2	0	3
6	26AEE02405T	**Professional core - VII** • AI Integrated Signal Processing and Systems	2	0	2	0	3
7	26AEE02401P	**Professional core - IV lab** • AI Assisted Control Systems Lab	0	0	0	3	1.5
8	26AEE02403P	**Professional core – V Lab** • AI-Assisted Control of Induction and Synchronous Machines Lab	0	0	0	3	1.5
9	26AEE02406S	**Skill Enhancement course - II** • Electrical Simulation Tools	1	0	0	2	2
10	26AHS0302M	Technical Paper Writing & IPR	2	0	0	0	0
		Total	**15**	**0**	**8**	**8**	**21**
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation							

I Year I Sem

Course Code	ENGINEERING CHEMISTRY FOR ELECTRICAL ENGINEERS	L	T	Pr	P	Credits
26AHS00105T		3	0	0	0	3

Course Objectives

1. Understand electrochemical energy storage and conversion for power and mobility applications.
2. Analyze corrosion mechanisms in electrical infrastructure and develop protection strategies.
3. Study electrical materials (conductors, insulators, semiconductors, magnetic materials) and their chemistry.
4. Evaluate coolants and insulating fluids for thermal management of electrical equipment.
5. Apply AI/ML tools for predictive maintenance, material selection, and energy optimization.

Course Content

UNIT I: Electrochemical Energy Storage and Conversion

Core Content: Fundamentals of electrochemistry: electrode potential, Nernst equation and problems, potentiometry (redox reactions), conductometry (strong acid vs strong base and weak acid vs strong base); electrochemical cells (Galvanic and electrolytic cells); primary batteries: construction, working and electrical applications of zinc-air and sodium-air batteries; secondary batteries: construction, charging-discharging reactions and applications of lead-acid, Ni-Cd, lithium-ion batteries; supercapacitors: definition, classification and applications; electric fuel cells: construction, working and electrical applications of hydrogen-oxygen and methanol-oxygen fuel cells; battery management systems: state of charge, thermal management, safety.

AI Integration: Battery health prediction (SoH/SoC) using regression/ML; anomaly detection for thermal runaway; optimization of charge–discharge profiles using data analytics.

Free/Open-Source AI Tools: Python (NumPy, Pandas, scikit-learn); Google Colab

UNIT II: Corrosion Chemistry and Protective Techniques

Core Content: Corrosion: definition, examples, types of corrosion — chemical corrosion and electrochemical corrosion, Pilling-Bedworth rule; galvanic corrosion, concentration cell corrosion, uniform, pitting, galvanic, crevice, stress and erosion corrosion; factors affecting corrosion; protective techniques (corrosion control): proper design, use of pure metal, metal alloys, inhibitors; cathodic protection — sacrificial anodic protection, impressed current cathodic protection; galvanizing: hot-dip galvanizing, electroplating; electroplating and electroless plating (nickel and copper); underground cable corrosion: stray current corrosion; corrosion of grounding systems: earth electrodes, grid corrosion; corrosion in electrical substations: steel structures, conductors; battery terminal corrosion; corrosion in cooling systems of electrical machines.

AI Integration: Predictive corrosion modeling using environmental/operational data; computer vision for coating/defect inspection; maintenance scheduling via ML.

Free/Open-Source AI Tools: Python (scikit-learn); OpenCV; Google Colab

UNIT III: Electrical Materials

Core Content: Introduction, classification and applications of conductors: copper, aluminium, silver, alloys (brass, bronze) — resistivity, conductivity, temperature coefficient; semiconductors: band theory, intrinsic and extrinsic semiconductors (Si, Ge, GaAs), doping — n-type (P, As) and p-type (B, Al); superconductors: Type I and Type II, BCS theory; insulators: definition, classification and dielectric properties — solid insulators (ceramics, porcelain, alumina, polymers, mica, glass), liquid insulators (transformer oil, synthetic esters, silicone fluids), gaseous insulators (air, SF_6, nitrogen); electrical polymers: types of polymerisation, thermoplastics and thermosetting plastics — preparation, properties and uses of Bakelite, polystyrene, polycarbonates, nylon-6, epoxy resins, butyl rubber, conducting polymers; magnetic materials: ferromagnetic, ferrimagnetic, paramagnetic.

AI Integration: Material selection using ML (property prediction, multi-objective optimization); defect classification in insulation using image analytics.

Free/Open-Source AI Tools: TensorFlow; Python; Orange Data Mining

UNIT IV: Coolants and Thermal Management in Electrical Systems

Core Content: Transformer oils: mineral oil, synthetic esters, natural esters, silicone fluids; properties of oils; oil testing: breakdown voltage, acidity, moisture content, dissolved gas analysis (DGA); coolants for electrical machines: air, hydrogen, water, oil; water cooling: deionized water, corrosion inhibitors; hydrogen cooling in large generators: properties, purity requirements, safety; cooling fluids for power electronics: water-glycol, dielectric fluids; nanofluids for enhanced heat transfer; fire-resistant fluids: less-flammable oils, synthetic fluids.

AI Integration: Predictive analytics for hot-spot temperature and insulation life; anomaly detection from DGA data; optimization of cooling systems using data-driven models.

Free/Open-Source AI Tools: Python (Pandas, SciPy); scikit-learn; Google Colab

UNIT V: Quantum Chemistry

Core Content: Fundamentals of quantum chemistry — black body radiation, photoelectric effect, wave-particle duality, De-Broglie wave equation, and Heisenberg's uncertainty principle; fundamentals of quantum mechanics, Schrodinger wave equation, significance of Ψ and Ψ^2; quantum numbers; particle in one-dimensional box; molecular orbital theory — bonding in homo- and hetero-nuclear diatomic molecules, energy level diagrams of O_2 and CO; π-molecular orbitals of butadiene and benzene, calculation of bond order.

AI Integration: Quantum system visualization and simple simulations; introduction to quantum-inspired models for materials.

Free/Open-Source AI Tools: Python (NumPy, SciPy); Google Colab

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level

CO1	Analyze electrochemical storage systems and select suitable technologies.	L3–L4
CO2	Evaluate corrosion in electrical infrastructure and design protection strategies.	L3–L5
CO3	Analyze and select electrical materials based on properties and applications.	L3–L4
CO4	Design thermal management solutions for electrical equipment.	L3–L5
CO5	Apply quantum chemistry concepts and AI tools for materials/energy analysis.	L3–L6

CO – PO Articulation Matrix

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	3	3	2	2	1	1	2	1	2	3	3	3
CO2	3	3	3	2	3	2	2	1	1	2	1	2	3	3	3
CO3	3	3	3	2	3	1	2	1	2	2	1	2	3	3	3
CO4	2	3	3	3	3	1	2	1	2	2	1	2	3	3	3
CO5	3	2	3	3	3	1	1	2	2	3	2	3	3	3	3

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.

Reference Books

1. Linden, D., & Reddy, T. B. (Eds.). (2001). Handbook of Batteries (3rd ed.). McGraw-Hill.
2. Roberge, P. R. (2008). Corrosion Engineering: Principles and Practice. McGraw-Hill.
3. Shikha, A., & Sharma, R. K. (2019). Engineering Chemistry. Krishna Prakashan Media.
4. Dara, S. S., & Umare, S. S. (2018). Engineering Chemistry (15th ed.). S. Chand Publishing.

Web Resources & NPTEL Links

1. IEEE Standards – https://standards.ieee.org/
2. IEC Standards for Electrical Equipment – https://www.iec.ch/
3. IRENA – Renewable Energy Technologies – https://www.irena.org/
4. U.S. Department of Energy – Energy Storage Database – https://www.energy.gov/eere/buildings/grid-interactive-efficient-buildings
5. Bureau of Indian Standards (BIS) – https://www.services.bis.gov.in/

Recommended Free & Open-Source AI/Computational Tools

1. Python (NumPy, Pandas, Matplotlib) – FREE on Google Colab
2. TensorFlow / scikit-learn – free, open-source

3. OpenCV – free, open-source
4. Orange Data Mining – free, open-source

I Year I Sem

Course Code	LINEAR ALGEBRA AND CALCULUS	L	T	Pr	P	Credits
26AHS00102T	Common to All Branches	3	0	2	0	4

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, eigenanalysis, and multivariable calculus to solve engineering problems.
- Use computational tools for approximation, optimization, differentiation, and integration.

Course Content

UNIT I: Matrix Algebra and Linear Systems

Core 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 ChatGPT/Copilot to explain row operations and rank reduction.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib) on Google Colab; SymPy for symbolic matrix operations; Jupyter Notebook; Scilab.

UNIT II: Eigenvalues, Eigenvectors, Diagonalization & Quadratic Forms

Core Content: Eigenvalues 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 eigenvalue computation and verification; Python/SciPy visualization of eigenmodes 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 III: Single Variable Calculus and Series Expansions

Core 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 for verification.

UNIT IV: Multivariable Differentiation and Optimization

Core 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; divergence and curl.

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 V: Multiple Integrals and Coordinate Transformations

Core 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.

Course Outcomes (COs)

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 eigenvalue 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)

CO – PO Articulation Matrix

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

CO	BL	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	L3	3	3	2	2	1	-	-	-	1	2	-
CO2	L4	3	3	3	2	1	-	-	-	1	2	-
CO3	L3	3	2	2	2	1	-	-	-	-	2	-
CO4	L4	3	3	3	3	2	-	-	-	1	2	-
CO5	L5	3	3	2	3	3	-	1	1	2	2	1

CO-PO Weightage

Average weightage based on the COs is shown below; it is aligned to NBA UG POs PO1 to PO11 and kept within the 1–3 correlation scale.

Textbooks

1. B. S. Grewal, Higher Engineering Mathematics, 44th Edition, Khanna Publishers, 2017.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, Wiley, 2018.
3. G. B. Thomas, M. D. Weir, J. Hass, Thomas' Calculus, 14th Edition, Pearson, 2018.
4. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, 5th Edition, Alpha Science International, 2021.

Reference Books

1. Gilbert Strang, Introduction to Linear Algebra, 5th Edition, Wellesley-Cambridge Press, 2016.
2. David C. Lay, Steven R. Lay, Judi J. McDonald, Linear Algebra and Its Applications, 5th Edition, Pearson, 2015.
3. George B. Arfken, Hans J. Weber, Frank E. Harris, Mathematical Methods for Physicists, 7th Edition, Academic Press, 2012.
4. Dennis G. Zill and Warren S. Wright, Advanced Engineering Mathematics, 6th Edition, Jones & Bartlett, 2018.

5. E. Kreyszig, Elementary Linear Algebra, 2nd Edition, Wiley, 1983.

Web Resources & NPTEL Links

1. NPTEL: Linear Algebra – https://nptel.ac.in
2. MIT OpenCourseWare – Linear Algebra: https://ocw.mit.edu
3. MIT OpenCourseWare – Multivariable Calculus: https://ocw.mit.edu
4. Python SymPy Documentation: https://docs.sympy.org
5. Google Colab for Python: https://colab.research.google.com
6. SciPy Documentation: https://docs.scipy.org
7. NumPy Documentation: https://numpy.org/doc

Recommended Free & Open-Source AI/Computational Tools

1. Python (NumPy, SciPy, Matplotlib) – FREE on Google Colab for matrix computation, visualization, and numerical analysis.
2. SymPy (free, open-source) – for symbolic algebra, calculus, matrix operations, and verification.
3. scikit-learn (free, open-source) – for regression, classification, and exploratory predictive modeling.
4. Jupyter Notebook (free, open-source) – for interactive computation and lab documentation.
5. OpenCV (free, open-source) – for image-based experimentation and visualization support where needed.
6. GNU Octave (free, open-source) – MATLAB-like numerical computing for matrix and calculus workflows.
7. TensorFlow / Keras (free, open-source) – for optional AI demonstrations in regression and classification tasks.

I Year I Sem

Course Code	DESIGN THINKING	L	T	Pr	P	Credits
26ACS05101T	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.

Course Content

UNIT I: Introduction to Design Thinking

Core Content:

- 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 II: Empathise and Define Stages

Core Content:

- 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 III: Ideation Stage

Core Content:

- 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 IV: Prototyping

Core Content:

- 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 V: Testing, Iteration and AI in Design

Core Content:

- 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: ChatGPT – 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 (COs)

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

CO – PO Articulation Matrix

Scale: 3 = High, 2 = Medium, 1 = Low, '-' = No correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	2	2	1	1	2	1	1	2	2	1	2	1	1	1
CO2	1	2	3	2	2	3	1	1	3	2	2	2	2	2	2

CO3	2	3	3	2	3	3	1	1	2	2	2	3	2	2	2
CO4	2	3	3	3	3	2	1	2	2	2	2	3	2	2	2
CO5	2	3	3	3	3	2	2	2	3	3	3	3	3	3	3

Textbooks

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

Reference Books

1. IDEO.org. (2015). The Field Guide to Human-Centered Design. IDEO.org, San Francisco.
2. Kelley, Tom & Kelley, David. (2013). Creative Confidence: Unleashing the Creative Potential Within Us All. Crown Business.
3. Liedtka, Jeanne & Ogilvie, Tim. (2011). Designing for Growth: A Design Thinking Tool Kit for Managers. Columbia Business School Press.
4. Plattner, Hasso, Meinel, Christoph & Leifer, Larry (Eds.). (2018). Design Thinking Research. Springer.
5. Kumar, Vijay. (2012). 101 Design Methods: A Structured Approach for Driving Innovation. Wiley.
6. Stickdorn, Marc & Schneider, Jakob. (2021). This Is Service Design Thinking (Expanded ed.). BIS Publishers.

Web Resources & NPTEL Links

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

Video Resources

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

I Year I Sem

Course Code	COMPUTATIONAL THINKING AND PROBLEM SOLVING WITH PYTHON	L	T	Pr	P	Credits
26ACS05102T	Common to All Branches (I Year I Sem)	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 Content

UNIT I: Computational Thinking and Programming Basics

Core 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); TB2 Chapters 1–2 (Thareja).

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.

Free/Open-Source AI Tools: Flowgorithm; Lucidchart; Algorithm Visualizer; ChatGPT

Unit Outcomes: Apply computational thinking — abstraction, decomposition, pattern recognition — to a given problem. Write algorithms and draw flowcharts for simple problems. Write and execute Python programs using variables, data types, and operators.

UNIT II: Decision Making and Looping

Core 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); TB1 Chapters 4–5 (Spronck).

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.

Free/Open-Source AI Tools: Python Tutor; Thonny; Replit; ChatGPT

Unit Outcomes: Write Python programs using if, if-else, and if-elif-else for decision making. Implement while and for loops with break, continue, and pass for iterative problems. Develop menu-driven and pattern programs using control statements.

UNIT III: Strings and Data Structures

Core 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); TB1 Chapters 6–7 (Spronck).

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.

Free/Open-Source AI Tools: NLTK; TextBlob; Pandas; NumPy; ChatGPT

Unit Outcomes: Apply string operations — reverse, palindrome check, frequency count. Implement list operations — sorting, searching, merging, nested lists. Apply set operations (union, intersection, difference) and tuple packing/unpacking.

UNIT IV: Functions and Problem Solving

Core 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); TB1 Chapters 8–9 (Spronck).

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.

Free/Open-Source AI Tools: Jupyter Notebook; SymPy; PyCharm; ChatGPT

Unit Outcomes: Write user-defined functions with all argument types and explain scope rules. Implement recursive functions for factorial, Fibonacci, and similar problems. Use lambda functions and dictionaries for compact, reusable Python programs.

UNIT V: File Handling, Exception Handling, and OOP

Core 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); TB1 Chapter 10 (Spronck).

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.

Free/Open-Source AI Tools: Pandas; OpenPyXL; Pytest; ChatGPT; Django

Unit Outcomes: Read and write text files; process CSV/Excel files using Pandas and OpenPyXL. Implement exception handling using try-except-finally for robust Python programs. Design and implement classes with constructors and methods using OOP principles.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Apply computational thinking concepts to design algorithms, flowcharts, and basic 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 for problem solving.	L4 (Analyze)
CO4	Design modular programs using functions, recursion, and dictionaries to improve code reusability.	L4 (Analyze)
CO5	Implement file handling, exception handling, and basic object-oriented programming concepts in Python applications.	L3 (Apply)

CO – PO Articulation Matrix

Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. (Note: this matrix was not provided in the source syllabus and has been constructed to align the COs of this course with the standard PO framework.)

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	-	-	-	-	-	-	1	2	2
CO2	3	3	2	1	2	-	-	-	-	-	-	1	2	2
CO3	2	3	2	2	2	-	-	-	-	1	-	1	2	2
CO4	2	3	3	2	2	-	-	-	1	1	-	2	2	2
CO5	2	2	3	2	3	-	-	1	1	1	-	2	2	2

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.

Web Resources & NPTEL Links

3. NPTEL – Problem Solving through Programming in C / Python – nptel.ac.in

Recommended Free & Open-Source AI/Computational Tools

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

I Year I Sem

Course Code	AI TOOLS AND APPLICATIONS	L	T	Pr	P	Credits
26ACS05103T	Common to All Branches (I Year I Sem)	1	0	4	0	3

Course Objectives

- Introduce students to AI concepts and the landscape of AI tools available for everyday academic and professional use.
- Develop practical skills in using AI for document creation, presentations, writing, and academic tasks applicable across all engineering disciplines.
- Build proficiency in AI-powered data analysis, spreadsheet automation, and intelligent research tools without requiring programming knowledge.
- Expose students to AI tools in visual design, media creation, and note-taking, and progressively to AI applications specific to their engineering branch.
- Cultivate responsible AI habits: verifying outputs, protecting privacy, respecting intellectual property, and understanding 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	Bloom's Taxonomy Level
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)	L3 (Apply)
CO2	Use AI tools to create documents and presentations; apply prompt engineering and NotebookLM to improve writing, research, and study habits. (L3)	L4 (Analyse)
CO3	Perform data analysis using AI without coding; use AI-powered research and note-taking tools to find, verify, organize, and summarize information. (L3)	L4 (Analyse)
CO4	Apply AI tools in visual design, diagrams, and media; identify and verify AI applications relevant to their own engineering discipline. (L4)	L5 (Evaluate)
CO5	Demonstrate responsible AI usage — bias, privacy, ethics, governance; design and present an AI-assisted mini-project using tools from all units. (L5)	L5 (Evaluate)

Course Content

UNIT – I	UNDERSTANDING AI AND AI FOR DOCUMENTS AND PRESENTATIONS	9 Hrs

Content	- 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 Copilot in PowerPoint — slide generation, design suggestions, speaker notes; Canva AI — Magic Design, text-to-presentation, brand kits; Beautiful.ai — smart slide layouts; comparing AI-generated vs. manually created presentations. - AI for document writing: Microsoft Copilot in Word — draft, rewrite, summarize; Gemini in Google Docs — writing assistance and inline Q&A; Notion AI — meeting notes, project plans, knowledge base; Claude Projects — organizing documents and having 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 GPTZero.
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

Content	- NotebookLM (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) from sources; 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 NotebookLM 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 Symbolab to solve and verify equations from your current semester subjects; use Copilot in Excel or ChatGPT ADA to analyse a real dataset.

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

Content	- 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 infographics 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.
AI Integration	Use Napkin.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 samples and analyse confidence scores.

UNIT – IV	**AI ACROSS ENGINEERING DISCIPLINES**	**9 Hrs**

Content	- 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 cybersecurity — anomaly detection 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

Content	- 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 5 daily-use AI systems using EU AI Act risk tiers; use GPTZero and Google Lens to detect AI-generated content; design and present the Capstone Mini-Project integrating tools from all five units.

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 / PSO Articulation Matrix (NBA)

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

CO	Bloom's Lvl	PO1 Eng. Know.	PO2 Problem Analysis	PO3 Design/ Dev.	PO4 Invest-igation	PO5 Modern Tools	PO6 Engineer & Society	PO7 Environ. & Sust.	PO8 Ethics	PO9 Team Work	PO10 Commun.	PO11 Prj Mgm
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.

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.
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 Year I Sem

Course Code	ENGINEERING WORKSHOP	L	T	Pr	P	Credits
26AME03101P	Common to All Branches (I Year I Sem)	0	0	0	3 (1.5)	undefined

Prerequisites: Basic Computing Elements

Course Objectives

- Create awareness on safety precautions required at workplace.
- Master fundamental hand tools and traditional manufacturing craftsmanship.
- Practice on manufacturing of components using workshop trades including fitting, sheet metal and carpentry, foundry and welding.
- Impart basic electrical engineering knowledge for House Wiring Practice.
- Demonstrate fundamentals of foundry and welding.

Course Content

1. Safety Practices & Precautions

Core Content: Demonstrate the necessity of workshop hazards, various Personal Protective Equipment and emergency protocols.

AI Integration: Computer vision for safety compliance.

2. Wood Working

Core Content: Introduction to different types of woods, carpentry tools, timber joints — (a) Half-Lap joint (b) Mortise and Tenon joint (c) Corner Dovetail joint or Bridle joint.

3. Fitting

Core Content: Familiarity with different types of tools used in fitting and perform fitting exercises like (a) V-fit (b) Dovetail fit (c) Semi-circular fit.

4. Sheet Metal Working

Core Content: Familiarity with different types of tools used in sheet metal working; developments of the following sheet metal jobs from GI sheets: (a) Tapered tray (b) Conical funnel (c) Elbow pipe (d) Brazing.

5. Electrical Wiring

Core Content: Introduction to electrical tools, safety precautions, wiring materials. Familiarity with different types of basic electrical circuits and making the following connections: (a) Parallel and series (b) Two-way switch (c) Godown lighting (d) Tube light (e) Three phase motor (f) Soldering of wires.

6. Plumbing

Core Content: Introduction to plumbing tools, pipe fittings, thread cutting, making a simple PVC pipe joint. Preparation of pipe joints with coupling for same diameter and with reducer for different diameters.

7. Welding Shop

Core Content: Demonstration and practice on Arc Welding and Gas welding. Preparation of a simple Lap joint/Butt joint.

8. Foundry Trade

Core Content: Demonstration and practice on moulding tools and processes; preparation of Green Sand Moulds for simple patterns. Introduction to 3D Printing: core concepts, hardware, software, digital file and 3D printing materials; familiarisation with the 3D printing process for complex objects. Computer Components: disassemble and reassemble a desktop computer to understand the role of each component (CPU, motherboard, RAM, HDD/SSD, GPU, etc.). Hardware Troubleshooting & Peripheral Installation: simulate common hardware issues (e.g., RAM failure, CPU overheating) and identify resolution procedures; install and configure peripheral devices such as printers, scanners, and external hard drives. Network Setup and Configuration: set up a small LAN network with routers, switches, and computers; configure IP addresses, subnet masks, and basic network services. Disassembly and Assembly of equipment such as bicycle, gear box, braking system: familiarise with disassembly and assembly process.

AI Integration: Smart energy monitoring using IoT current sensors; building machine learning load forecasting models; training anomaly detection systems to automatically flag electrical arc or leakage patterns. Develop predictive load-monitoring models that detect faulty configurations or phantom energy drains.

Course Outcomes (COs)

CO	Course Outcome Statement
CO1	Recognize the importance of safety precautions while working in the workshop.
CO2	Identify different measuring, marking, cutting tools used in the workshop and their operational capabilities.
CO3	Fabricate precision wood and metal fitting joints within standard specifications.
CO4	Understand wire standards, safe residential switches, lighting circuits, and basic appliances.
CO5	Identify different tools used in foundry and welding sections.

CO – PO Articulation Matrix

Scale: 1 (Low/Slight), 2 (Medium/Moderate), and 3 (High/Substantial)

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	-	-	-	2	-	-	-	-	-	-	2	2	2
CO2	3	2	2	1	2	-	-	-	-	-	-	3	2	1

CO3	3	1	-	-	1	2	-	-	-	-	-	2	2	1
CO4	-	-	-	-	-	-	-	1	3	-	1	3	2	2
CO5	-	-	-	-	-	-	-	-	-	3	-	3	2	2

Textbooks

1. Workshop Technology Vol I & II, S.K. Hajra Choudhury, Media Promoters & Publishers.
2. A Course in Workshop Technology, B.S. Raghuwanshi, Dhanpat Rai & Co., 2015 & 2017.
3. Hands-On Industrial Internet of Things, 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 Edition, 2015.
5. Wiring Estimating, Costing and Contracting, Soni P.M. & Upadhyay P.A., Atul Prakashan, 2021–22.

Reference Books

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

Web Resources & NPTEL Links

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

I Year I Sem

Course Code	ENGINEERING CHEMISTRY FOR ELECTRICAL ENGINEERS LAB	L	T	Pr	P	Credits
26AHS00105P	I Year I Sem	0	0	0	3	1.5

Course Content (List of Experiments)

1. Estimation of copper using sodium thiosulphate (iodometry).
2. Determination of acid strength in Ni-Cd battery.
3. Preparation of polymer (Bakelite).
4. Estimation of ferrous iron (Fe^{2+}) using potassium dichromate (K2Cr2O7) (redox titrations).
5. Determination of corrosion.
6. Determination of corrosion by environment.
7. Conductometric titrations of strong acid vs strong base (e.g., HCl vs NaOH).
8. Conductometric titrations of weak acid vs strong base (e.g., CH3COOH vs NaOH).
9. Potentiometry (redox titrations) — estimation of iron (Fe^{2+}) using potassium dichromate (K2Cr2O7).
10. Determination of viscosity of lubricating oil using Red-Wood Viscometer – I.
11. Determination of viscosity of lubricating oil using Red-Wood Viscometer – II.

AI Integration

AI Extension: Data logging and analysis using Python/Excel; predictive models for corrosion rates, battery parameters, and lubricant properties.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Recall and explain the principles of volumetric, conductometric, potentiometric, and corrosion experiments used in engineering chemistry laboratory.	BL-1 (Remember), BL-2 (Understand)
CO2	Perform chemical experiments for estimation of metals, battery parameters, polymer preparation, and viscosity measurements by following standard laboratory procedures.	BL-3 (Apply)
CO3	Analyze experimental data obtained from titrations, corrosion studies, and lubricant characterization to determine chemical parameters and interpret results.	BL-4 (Analyze)
CO4	Apply computational and AI-based tools such as Python and Excel for data logging, visualization, and prediction of corrosion rates, battery characteristics, and lubricant	BL-3 (Apply), BL-4 (Analyze)

	properties.	
CO5	Demonstrate professional laboratory practices, teamwork, safety standards, and effective communication while conducting chemistry experiments and preparing reports.	BL-3 (Apply)

CO – PO Articulation Matrix

Scale: 1 (Low/Slight), 2 (Medium/Moderate), and 3 (High/Substantial)

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	2	2	2
CO2	2	3	-	2	1	-	-	2	2	1	-	3	2	1
CO3	2	3	-	3	2	-	-	-	-	1	-	2	2	1
CO4	1	2	2	3	3	-	-	-	-	1	1	3	2	2
CO5	-	-	-	-	-	-	-	3	3	3	2	3	2	2

Web Resources & NPTEL Links

1. IEEE Standards
2. IEC
3. IRENA
4. U.S. DOE (Energy Storage)
5. BIS

Recommended Free & Open-Source AI/Computational Tools

1. Python (NumPy, Pandas, Matplotlib)
2. TensorFlow / scikit-learn
3. OpenCV
4. Orange Data Mining

I Year I Sem

Course Code	COMPUTATIONAL THINKING AND PROBLEM SOLVING WITH PYTHON LAB	L	T	Pr	P	Credits
26ACS05102P	Common to All Branches	0	0	0	3	1.5

Lab Course Outcomes

- Design algorithms and flowcharts for simple problems; write and execute basic Python programs using variables, data types, and operators.
- Implement programs using if-else, nested if, switch (match), while, and for loops to solve decision-making and iterative problems.
- Apply string operations and Python built-in data structures — lists, tuples, and sets — to solve data manipulation problems.
- Write modular programs using user-defined functions, recursion, lambda expressions, and dictionaries.
- Implement programs using file handling, exception handling, and basic object-oriented programming concepts in Python.

List of Experiments

Unit I – Computational Thinking and Programming Basics

1. **Algorithm Design, Flowcharts, and Python Basics**
   Draw flowcharts using Flowgorithm for three problems: (a) find the largest of two numbers; (b) check whether a number is even or odd; (c) compute the factorial of a given number. For each flowchart, trace through the logic manually with 3 sample inputs before coding. Write equivalent Python programs for all three; run them in Thonny or Google Colab and verify that the output matches the manual trace. Use Python Tutor to visualize the variable state at each step for the factorial program.
2. **Data Types, Operators, and Type Conversion**
   Write Python programs to: (a) declare variables of all primitive data types (int, float, complex, bool, str) and print their type using type(); (b) demonstrate all operator categories — arithmetic (+, −, ×, ÷, //, %, **), relational (==, !=, <, >, <=, >=), logical (and, or, not), bitwise (&, |, ^, ~, <<, >>), and assignment operators — with one example each; (c) demonstrate implicit and explicit type conversion — int to float, float to int, int to str, str to int using int(), float(), str(); (d) print a formatted bill: accept item name, quantity, and price as input; compute and display total using f-string formatting with two decimal places.
3. **Input, Output, and Expression Evaluation**
   Write Python programs to: (a) accept the radius of a circle from the user; compute and display area (π r^2) and circumference (2π r) using the math module; (b) accept three subject marks; compute total, percentage, and grade (O ≥ 90, A+ ≥ 80, A ≥ 70, B+ ≥ 60, B ≥ 50, C ≥ 40, F < 40); print a formatted result card using print() with appropriate spacing; (c) accept two numbers and an operator (+, −, ×, ÷) as input; evaluate the expression using eval(); handle division by zero using a simple if condition; (d) demonstrate operator precedence: compute and print results of 5 mixed expressions — predict the result manually first, then verify by running the program.

Unit II – Decision Making and Looping

4. **Decision Control — if, elif, else, and match**

Write Python programs to: (a) accept a year and check whether it is a leap year using nested if — a year is a leap year if divisible by 400, or divisible by 4 but not by 100; test with 2000, 1900, 2024, 2023; (b) accept a character and classify it as uppercase letter, lowercase letter, digit, or special character using if-elif-else; (c) accept a day number (1–7) and print the day name using match-case (Python 3.10+); add a default case for invalid input; (d) implement a simple ATM menu using nested if — display options (Check Balance, Deposit, Withdraw, Exit); perform the selected operation; display appropriate error messages for invalid PIN or insufficient balance.

5. **Loops — while, for, and Nested Loops**

   Write Python programs to: (a) print multiplication table of a number entered by the user using a for loop (1 to 10); (b) print all prime numbers between 1 and 100 using a while loop with a flag variable — also count and display the total number of primes found; (c) print the following five patterns using nested for loops: right-angled triangle (stars), inverted triangle, number pyramid, Floyd's triangle, and a diamond pattern — accept the number of rows as input for each; (d) implement a number guessing game — generate a random number between 1 and 50 using random.randint(); allow the user a maximum of 7 attempts; print 'Too High' or 'Too Low' as hints; use break when the correct number is guessed; display attempts used.

6. **Loop Control and Application Programs**

   Write Python programs to: (a) compute the sum of digits, reverse, and check if palindrome for a number entered by the user — use a while loop extracting digits using % and //; test with 121, 12321, 1234; (b) compute the sum of the Fibonacci series up to N terms using a for loop; also print the series itself; (c) accept a positive integer N and compute: sum of natural numbers (1+2+…+N), sum of squares ($1^2+2^2+\ldots+N^2$), and sum of cubes ($1^3+2^3+\ldots+N^3$) — verify sum of cubes formula: $[N(N+1)/2]^2$; (d) implement a simple calculator using a while loop with a menu — loop continues until the user selects Exit; use continue to handle invalid menu choices without breaking the loop.

Unit III – Strings and Data Structures

7. **String Operations and Methods**

   Write Python programs to: (a) accept a string and perform: count vowels and consonants, reverse the string, check palindrome, count words, convert to title case — display all results; (b) demonstrate 10 built-in string methods: upper(), lower(), strip(), replace(), split(), join(), find(), count(), startswith(), endswith() — with one practical example each; (c) accept a sentence and: find the longest word, count frequency of each word (store in a dictionary), print words in alphabetical order; (d) implement a simple Caesar cipher — accept a message and a shift value (1–25); encrypt each letter by shifting it; display encrypted message; write a decrypt function to recover the original message.

8. **Lists and Tuples**

   Write Python programs to: (a) create a list of 10 integers; perform: insert at a given position, delete by value, find maximum and minimum without using built-in max/min, sort using Bubble Sort, search using linear search — display the list after each operation; (b) demonstrate list slicing: extract first half, second half, every alternate element, and reverse using slice notation; compare with string slicing on the same index values; (c) store student records as a list of tuples — each tuple: (roll_no, name, marks); sort by marks (descending) using sorted() with a lambda key; print rank list; find the student with highest marks using max() with a key; (d) demonstrate list comprehension: generate squares of even numbers from 1 to 20; filter words longer than 4 characters from a given word list; create a multiplication table as a 2D list using nested comprehension.

9. **Sets and Applications**

Write Python programs to: (a) create two sets A and B from user input; compute and display: union (A | B), intersection (A & B), difference (A − B and B − A), symmetric difference (A ^ B); verify De Morgan's laws: not(A | B) == (not A) & (not B) for sample values; (b) use a set to remove duplicates from a list of 20 student roll numbers (with deliberate duplicates); compare list size before and after; print the unique roll numbers in sorted order using sorted(set()); (c) demonstrate frozen sets: create a frozenset of prime numbers up to 20; attempt to add an element (show TypeError); use it as a dictionary key (demonstrate immutability advantage); (d) implement a program to find: students who passed all three subjects, students who failed at least one, students who passed exactly two — store each class in a set and use set operations.

Unit IV – Functions and Problem Solving

10. Dictionaries and Applications

Write Python programs to: (a) create a student marks dictionary {name: [marks_list]}; compute and display average marks, highest scorer, and students who scored below 50; add a new student, update an existing student's marks, delete a student — show the dictionary after each operation; (b) implement a word frequency counter — accept a paragraph as input; split into words; build a frequency dictionary using a for loop; print the top 5 most frequent words sorted by count descending; (c) demonstrate all dictionary methods: keys(), values(), items(), get(), update(), pop(), setdefault(), copy() — with one example each; (d) build a simple phone book — store contacts as {name: phone}; support: add contact, search by name, delete contact, display all contacts sorted alphabetically — use a while loop menu.

11. User-Defined Functions and Scope

Write Python programs to: (a) write 5 utility functions: is_prime(n), is_palindrome(s), factorial(n), gcd(a,b), power(base, exp) — call each with 3 test inputs and verify results; (b) demonstrate all four argument types: positional (def greet(name, age)), default (def greet(name, age=18)), keyword (greet(age=20, name='Ram')), and variable-length (*args for sum of any count, **kwargs for student info display) — write one function for each and test; (c) demonstrate local vs. global scope: write a program with a global counter variable that is modified inside a function using the global keyword; contrast with a local variable of the same name — print both; (d) write a function find_stats(numbers) that accepts a list and returns mean, median, mode, and standard deviation without using the statistics module — test on [4, 7, 13, 2, 7, 4, 7, 1].

12. Recursion and Lambda Functions

Write Python programs to: (a) implement recursive functions for: factorial(n), fibonacci(n), sum_of_digits(n), binary_search(arr, target, low, high), and power(base, exp) — for each, use Python Tutor to trace the call stack for a small input and count the total recursive calls; (b) compare iterative vs. recursive factorial for n = 5, 10, 15, 20 on execution time using time.time(); document when recursion becomes slower; (c) implement lambda functions for: square, cube, even/odd check, Celsius-to-Fahrenheit, and concatenate two strings — use each with map() and filter() on a list of 10 elements; (d) write a higher-order function apply_twice(f, x) that applies function f to x twice; test with square, double, and increment functions; demonstrate that lambda functions work as arguments.

Unit V – File Handling, Exception Handling, and OOP

13. File Handling

Write Python programs to: (a) create a text file and write 10 lines of student records (roll number, name, marks); read and display all lines; append 5 more records; count total lines; display only records where marks > 60; (b) copy the contents of one text file to another; count total characters, words, and lines in the source file; find and replace a specific word (e.g., 'Python' →

'programming') and write the result to a new file; (c) read a CSV file (student data) using the csv module; compute average marks per subject; write a summary report CSV with student name and grade; (d) create a student marks file; sort all records by marks in descending order (read all, sort in memory, rewrite); demonstrate the difference between read modes: 'r', 'w', 'a', 'r+' with examples.

14. Exception Handling

Write Python programs to: (a) write a program that accepts two numbers and performs division; handle ZeroDivisionError, ValueError (non-numeric input), and TypeError in separate except blocks; use a finally block to print 'Operation complete' regardless of outcome; (b) implement a robust file reader that handles FileNotFoundError and PermissionError; use try-with multiple except clauses; after handling each exception, print a meaningful user-friendly error message rather than a traceback; (c) define a custom exception class InsufficientFundsError(Exception) with a message attribute; implement a BankAccount class with deposit() and withdraw() methods — withdraw() raises InsufficientFundsError if balance is insufficient; test with deposits and withdrawals; (d) demonstrate exception chaining (raise X from Y) — catch a ValueError and re-raise as a custom ApplicationError with the original exception attached; use raise without arguments in a re-raise scenario.

15. Object-Oriented Programming — Class Design Capstone

Design and implement a Student Management System using OOP: (a) create a Student class with attributes: roll_no, name, marks (list of 5 subjects) and methods: compute_percentage(), get_grade() (O/A+/A/B+/B/C/F), display() — create 5 Student objects with different data; (b) add a class variable total_students that increments with each object creation; add a class method get_total() and a static method validate_marks(marks) that returns True if all marks are between 0 and 100; (c) demonstrate object interactions: write a find_topper(students) function that accepts a list of Student objects and returns the one with the highest percentage; sort the list of Student objects by percentage using sorted() with a lambda key on obj.compute_percentage(); (d) write all student records to a file using the write() method; read them back and reconstruct Student objects; print the final class report in tabular format using f-strings with column alignment.

AI Tools Integration (Lab): Use Thonny IDE (beginner-friendly, built-in debugger) or Google Colab as the primary lab environment. Use Python Tutor (pythontutor.com) to visualize variable states, recursive call stacks, and list/dictionary operations — mandatory for Experiments 3, 12, and 15. Use Flowgorithm to design flowcharts before writing code in Experiments 1 and 4. Use ChatGPT or Claude to explain error messages when a program fails — read the explanation, fix the error yourself, and document both the error and the fix. Do NOT copy AI-generated code directly; use AI to understand concepts and debug, then write the solution independently.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Explain computational thinking concepts and develop algorithms, flowcharts, and basic Python programs using variables, data types, operators, and expressions.	BL1, BL2

CO2	Develop Python programs using decision-making statements, loops, strings, and built-in data structures to solve computational problems.	BL3 (Apply)
CO3	Design modular Python programs using functions, recursion, lambda expressions, and dictionaries for problem solving and data processing applications.	BL3 (Apply)
CO4	Analyze and implement file handling, exception handling, and object-oriented programming concepts to develop robust Python applications.	BL4 (Analyze)
CO5	Utilize modern programming tools and debugging environments such as Thonny, Google Colab, Python Tutor, and AI-assisted debugging tools for program development and testing.	BL3, BL4

CO – PO Articulation Matrix

Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. (Note: this matrix was not provided in the source syllabus and has been constructed to align the COs of this course with the standard PO framework.)

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	-	-	-	1	2	2
CO2	3	3	2	2	2	-	-	-	-	-	-	1	2	2
CO3	2	3	3	2	2	-	-	-	1	1	-	1	2	2
CO4	2	3	3	3	2	-	-	1	1	1	-	2	2	2
CO5	1	2	1	2	3	-	-	1	2	2	-	2	2	2

LEARNING RESOURCES

Textbooks

1. TB1: Pieter Spronck, Computational Thinking with Python (course textbook — algorithms and Python foundations).
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. (Free at greenteapress.com)
2. Eric Matthes, Python Crash Course, 3rd Edition, No Starch Press.

Online Tools and Resources

- Python Tutor – pythontutor.com |Flowgorithm – flowgorithm.org
- Thonny IDE – thonny.org | Google Colab – colab.research.google.com
- Python Official Documentation – docs.python.org/3
- NPTEL – Problem Solving through Programming in Python – nptel.ac.in
- W3Schools Python Tutorial – w3schools.com/python |Replit – replit.com

I Year I Sem

Course Code	NSS / NCC / SCOUTS & GUIDES / COMMUNITY SERVICE	L	T	Pr	P	Credits
26AHS00101L	Common to All Branches (I Year I Sem)	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 Content

UNIT I: Foundation of NSS, NCC, Scouts & Guides and Community Service

Core Content: 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 Outcomes: Recall the history, founding philosophy and motto of NSS, NCC, and Scouts & Guides. Explain the organisational structure, ranks and regulatory framework of NSS and NCC. Describe the concept and significance of volunteerism and civic responsibility in a democratic society. Distinguish between different types of community service and identify relevant national programmes.

UNIT II: Leadership, Skill Development, Camping and Disaster Management

Core Content: 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 Outcomes: Apply leadership principles and team-building strategies in group activities and camp scenarios. Demonstrate correct parade, drill and basic physical training techniques. Organise and simulate camp logistics including safety protocols and adventure activity planning. Perform essential first-aid procedures (CPR, wound care, fracture management) on simulated casualties. Evaluate disaster risk and assign appropriate volunteer roles during simulated disaster scenarios.

UNIT III: Community Projects, Social Awareness and Service Documentation

Core Content: 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 Outcomes: Conduct a community needs assessment using participatory tools. Design a community service project plan with SMART goals, timelines and resource allocation. Execute awareness drives on health, environment, gender and digital literacy. Prepare comprehensive field reports and reflective portfolios documenting service outcomes. Evaluate the social impact of community service interventions using evidence.

AI-Integrated Activities

- AI-Assisted Community Needs Survey: use ChatGPT / Gemini to design survey questionnaires; analyse crowd-sourced responses with AI tools to map priority community issues.
- Virtual Camp Planning Simulator: use Notion AI / Trello AI to simulate camp logistics, assign roles and generate contingency plans for adverse scenarios.
- Disaster-Risk FAQ Chatbot: students build a first-aid guidance chatbot using Dialogflow / Botpress and deploy it for community awareness.
- AI Social Media Campaign Design: use Canva AI / Adobe Firefly to create Swachh Bharat / blood-donation awareness posters; schedule with Buffer AI.
- Reflective Portfolio with AI Writing Aid: draft field reports using Grammarly / ChatGPT; critically compare AI-generated drafts with self-written reflections.

Course Outcomes (COs)

CO	Course Outcome Statement	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)

CO – PO Articulation Matrix

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	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	-	-	-	-	-

Textbooks

- NSS Training Manual, Ministry of Youth Affairs & Sports, Govt. of India, MYAS Publication, 2022.
- National Cadet Corps – Training Syllabus & Manual, Directorate General NCC, DG NCC, New Delhi, 2021.

Reference Books

- Dr. S. K. Lal, Community Development & Social Service, Regal Publications, 2019.
- Satish Modh, Disaster Management – A Holistic Approach, Macmillan India, 2020.
- Steve McCurley & Rick Lynch, Volunteer Management, Energize Publications, 2018.
- Lord Robert Baden-Powell, Scouting for Boys, Oxford University Press (Centenary Ed.), 2008.

Web Resources & NPTEL Links

- 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
- 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

Recommended Free & Open-Source AI/Computational Tools

- ChatGPT / Gemini – for survey design and response analysis.
- Notion AI / Trello AI – for camp planning simulation.
- Dialogflow / Botpress (free tiers) – for chatbot development.
- Canva AI / Adobe Firefly – for awareness poster design.
- Grammarly / ChatGPT – for portfolio and field-report writing assistance.

I Year II Sem

Course Code	COMMUNICATIVE ENGLISH	L	T	Pr	P	Credits
26AHS00201T	Common to All Branches (I Year II Sem)	2	0	0	0	2

Course Objectives

- Recall and list fundamental concepts of English grammar, vocabulary, phonetics, and communication models essential for professional contexts.
- Explain and interpret various forms of written and oral communication including emails, reports, presentations, and group discussions.
- Use appropriate AI tools and digital platforms to enhance communicative accuracy, fluency, and professional writing skills in real-world scenarios.
- Examine and differentiate effective versus ineffective communication strategies in professional, academic, and intercultural contexts.
- Appraise and critique written and spoken communication samples using AI-assisted feedback mechanisms and peer-review methods.
- Design and produce professional-grade communication artifacts — including portfolios, reports, and presentations — integrating AI tools for quality enhancement.

Course Content

UNIT I: Foundations of English Communication

Weeks 1–3 | 6 Contact Hours

Core 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: Use of ChatGPT for grammar explanation, error identification, and instant Q&A on language rules; Grammarly for real-time grammar, spelling, and punctuation correction; Google Speech-to-Text for phonetics practice and pronunciation feedback; Merriam-Webster Vocabulary Builder for word-of-the-day, etymology, and interactive vocabulary quizzes. Hands-on activity: students paste a 100-word paragraph into Grammarly and use ChatGPT to understand each flagged error, then record their voice reading the corrected paragraph using Google Speech-to-Text and submit the original paragraph, AI-corrected version with error report, and a reflection note.

Free/Open-Source AI Tools: ChatGPT; Grammarly; Google Speech-to-Text; Merriam-Webster Vocabulary Builder

Unit Outcomes: Recall the fundamental components and types of the communication process. Identify and classify parts of speech and tense forms in given sentences. Explain sentence structure patterns and apply them in original writing samples. Articulate words with correct stress and intonation using digital phonetics tools.

UNIT II: Reading Comprehension & Writing Skills

Weeks 4–6 | 6 Contact Hours

Core 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 & blogs.

AI Integration: Use of QuillBot for AI paraphrasing of passages at varying complexity levels; Hemingway Editor for readability scoring, passive voice detection, and sentence simplification; ChatGPT for essay outline generation, argument structuring, and content brainstorming; Rewordify for vocabulary simplification. Hands-on activity: students upload a 300-word passage to QuillBot and generate three paraphrased versions, then revise their own essay in Hemingway Editor until readability reaches Grade 8 or below, submitting the original passage, paraphrased versions, and before/after readability scores.

Free/Open-Source AI Tools: QuillBot; Hemingway Editor; ChatGPT; Rewordify

Unit Outcomes: Demonstrate skimming and scanning strategies to extract key information from texts. Summarize and paraphrase academic passages using AI paraphrasing tools accurately. Compose well-structured paragraphs and short essays on technical and social topics. Evaluate the clarity and readability of own writing using AI-based readability scoring tools.

UNIT III: Professional & Business Communication

Weeks 7–9 | 6 Contact Hours

Core Content: Email writing and letter writing: formal & informal; short report writing: structure, types (formal/informal), and format; note-making and summarization techniques; resume/CV writing and LinkedIn profile optimization.

AI Integration: Use of ChatGPT to draft and refine professional emails, cover letters, reports, and proposals; Kickresume/Resume.io for AI-powered resume building with ATS-optimized templates; Notion AI for report structuring and knowledge management; Grammarly Business for tone detection and formality checks. Hands-on activity: each student drafts a formal job-application email and a one-page resume using ChatGPT and Kickresume, runs both through Grammarly Business for tone and professionalism checks, and conducts peer evaluation using a structured rubric.

Free/Open-Source AI Tools: ChatGPT; Kickresume / Resume.io; Notion AI; Grammarly Business

Unit Outcomes: Draft formal and semi-formal business emails and letters using professional writing conventions. Produce structured reports, proposals, and technical documents for workplace contexts. Design professional resumes and LinkedIn profiles using AI-assisted content generation tools. Analyze the structure, register, and tone of professional documents for accuracy and appropriateness.

UNIT IV: Listening and Speaking Skills

Weeks 10–12 | 6 Contact Hours

Core 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: Use of Gamma.app/Beautiful.ai for AI-powered presentation design; Orai (AI Speech Coach) for fluency analysis, filler-word detection, pacing, and confidence scoring; Otter.ai for auto-transcription and keyword analysis; Canva AI for visual communication design. Hands-on activity: students prepare a 5-minute presentation using Gamma.app or Canva AI, practice with Orai across at least 3 logged sessions, deliver the presentation in class, get it transcribed via Otter.ai, and submit slides, Orai screenshots, transcript, and peer-evaluation sheet.

Free/Open-Source AI Tools: Gamma.app / Beautiful.ai; Orai (AI Speech Coach); Otter.ai; Canva AI

Unit Outcomes: Demonstrate effective speaking skills in structured group discussions and mock interview settings. Differentiate between effective and ineffective presentation strategies using AI-generated feedback. Construct and deliver a well-organized 5-minute presentation using AI-designed slides. Evaluate peer presentations using AI-powered transcription tools and structured assessment rubrics.

UNIT V: Use of AI in Language Learning

Weeks 13–16 | 8 Contact Hours

Core Content: AI chatbots and virtual assistants in education: use of AI in grammar correction and vocabulary building, AI-based speaking and pronunciation tools, writing assistants, e-portfolio creation; AI in global communication: automated translation, summarizing large texts, and finding information in different languages (e.g., Google Translate, Reverso); 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: Use of Copyleaks/Turnitin for plagiarism detection and academic integrity analysis; DeepL Translator for high-accuracy translation; LinkedIn Learning/Coursera AI-literacy modules for understanding AI bias, ethics, and responsible usage; Wix/Adobe Portfolio for e-portfolio creation. Hands-on activity: students create a personal e-portfolio documenting 5 communication artifacts from all five units, each with an AI tool feedback snapshot, presented in the final class, peer-reviewed, and checked for plagiarism via Copyleaks, along with a 300-word reflection on AI tool usage across the course.

Free/Open-Source AI Tools: Copyleaks / Turnitin; DeepL Translator; LinkedIn Learning / Coursera; Wix / Adobe Portfolio

Unit Outcomes: Understand the role of AI in modern communication and develop awareness of technology-driven language use. Learn to use AI responsibly for improving speaking, writing, reading, and listening skills. Develop ethical awareness, critical thinking, and responsible digital communication habits. Improve professional communication and employability skills using modern AI tools.

Course Outcomes (COs)

Code	Bloom's Level	Statement
CO1	Remember	Students will be able to identify grammatical structures, phonetic patterns, and professional vocabulary, and recall communication

		frameworks using AI-assisted learning tools.
CO2	Understand	Students will be able to comprehend and interpret written and spoken communication models, and summarize content using AI paraphrasing and summarization tools.
CO3	Apply	Students will be able to compose professional emails, reports, and presentations using AI writing assistants such as ChatGPT, Grammarly, and Hemingway Editor.
CO4	Analyze	Students will be able to analyze discourse structures, identify communication barriers, and use AI-driven sentiment analysis tools to evaluate message effectiveness.
CO5	Evaluate, Create	Students will be able to evaluate communication samples using AI grammar checkers, readability scorers, and peer-evaluation rubrics aligned to professional standards. Students will be able to design and deliver professional communication artifacts — presentations, e-portfolios, and research reports — integrating AI tools for structure, style, and refinement.

CO – PO Articulation Matrix

Mapping Scale: 3 = Strong/High Correlation | 2 = Moderate Correlation | 1 = Weak/Indirect Correlation | – = No Direct Correlation

CO	BL	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	Remember	1	-	-	1	-	1	-	1	1	3	1	2	3	2
CO2	Understand	1	-	-	1	1	2	1	1	2	3	1	2	3	2
CO3	Apply	-	-	1	-	3	2	1	2	3	3	2	2	3	2
CO4	Analyze	-	1	1	2	2	2	1	2	2	3	2	2	3	2
CO5	Evaluate	-	1	1	2	3	2	2	3	2	3	2	3	3	2

Overall AI Tools Used in the Course

- ChatGPT (OpenAI) – https://chat.openai.com – Grammar, email drafting, essay structuring, content summarization
- Grammarly – https://www.grammarly.com – Real-time grammar, spelling, tone detection, and writing suggestions
- QuillBot – https://quillbot.com – Paraphrasing, summarization, and sentence restructuring
- Hemingway Editor – https://hemingwayapp.com – Readability scoring and writing clarity improvement

- Gamma.app – https://gamma.app – AI-powered presentation, document, and webpage creation
- Orai (Speech Coach) – https://www.orai.com – AI speech coaching – fluency, pacing, filler-word analysis
- Otter.ai – https://otter.ai – Auto-transcription and analysis of spoken communication
- Canva AI – https://www.canva.com – Visual communication, poster, infographic, and slide design
- Kickresume / Resume.io – https://www.kickresume.com – AI-assisted resume and cover letter generation
- Copyleaks / Turnitin – https://copyleaks.com – Plagiarism detection and academic integrity checking
- DeepL Translator – https://www.deepl.com – High-accuracy translation for intercultural communication
- Beautiful.ai – https://www.beautiful.ai – AI-driven professional slide deck design and layout
- Google Speech-to-Text – https://cloud.google.com/speech-to-text – Voice recognition for pronunciation and phonetics practice
- Notion AI – https://www.notion.so/product/ai – Note-taking, report structuring, and knowledge organization

Textbooks

- Meenakshi Raman & Sangeetha Sharma, Technical Communication: Principles and Practice, Oxford University Press, 4th Edition, 2022.
- Jyoti Shetty & Rajesh Pareek, Communicative English for Engineers and Professionals, Wiley India, 3rd Edition, 2021.

Reference Books

- Rizvi, M. Ashraf, Effective Technical Communication, Tata McGraw-Hill, 2nd Edition, 2018.
- Murphy, Herta A., Effective Business Communication, McGraw-Hill Education, 7th Edition, 2019.
- Guffey, Mary Ellen & Dana Loewy, Essentials of Business Communication, Cengage Learning, 11th Edition, 2021.
- Sethi, Anjanee & Bhavana Adhikari, Business Communication, Tata McGraw-Hill, 3rd Edition, 2020.
- Wren, P.C. & Martin, H., High School English Grammar and Composition, S. Chand, Revised Edition, 2019.
- Tickoo, M.L. & Suresh Kumar, E.R., Writing with a Purpose, Oxford University Press, 6th Edition, 2020.

Web Resources & NPTEL Links

- British Council – Learn English – https://learnenglish.britishcouncil.org
- Coursera – Communication Skills – https://www.coursera.org/courses?query=communication
- edX – Writing and Communication – https://www.edx.org/learn/writing
- Grammarly Blog – https://www.grammarly.com/blog
- NPTEL – Communication Skills – https://nptel.ac.in/courses/109104101
- BBC Learning English – https://www.bbc.co.uk/learningenglish

- Purdue OWL (Online Writing Lab) – https://owl.purdue.edu
- TED-Ed: How to Use a Comma – https://youtu.be/GHnl1O3NGwk
- English Speaking Course (Spoken English Guru) – https://youtu.be/1OECRXfM8RU
- How to Write a Professional Email – https://youtu.be/YcqhP_Cv5og
- Public Speaking Tips – TED Talk – https://youtu.be/iIBMbCFBVN8
- NPTEL Communication Skills – IIT Kharagpur – https://youtu.be/1Ck4k6rETBo
- How to Write a Resume (Step-by-Step) – https://youtu.be/Tt08KmFfIYQ
- Intercultural Communication Explained – https://youtu.be/YMyofREc5Jk
- Introduction to AI Writing Tools – https://youtu.be/p_Wq5WNzJcc

I Year II Sem

Course Code	DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS	L	T	Pr	P	Credits
26AHS00203T	Common to All Branches (I Year II Sem)	3	0	2	0	4

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 Content

UNIT I: First-Order Ordinary Differential Equations

Core 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 with initial conditions.

Free/Open-Source AI Tools: Python (SymPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; Wolfram Alpha free tier for solution verification; GNU Octave for ODE routines.

UNIT II: Higher-Order Linear Differential Equations

Core 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 and verification using SymPy; using AI tools to explain transient vs. steady-state response, resonance and Q-factor interpretation.

Free/Open-Source AI Tools: Python (NumPy, SciPy, SymPy, Matplotlib) on Google Colab; Jupyter Notebook; GNU Octave / MATLAB (optional) for LCR and mechanical system simulation; WolframAlpha free tier.

UNIT III: Partial Differential Equations

Core 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; prompting AI to classify mixed-derivative PDEs and student verification of AI reasoning.

Free/Open-Source AI Tools: Python (SymPy, NumPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; WolframAlpha free tier for PDE classification verification; Scilab.

UNIT IV: Vector Differentiation

Core 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; using AI tools to interpret Maxwell's equations in terms of divergence and curl operators; AI-assisted explanation of irrotational and solenoidal field classification.

Free/Open-Source AI Tools: Python (NumPy, SymPy, Matplotlib, mpl_toolkits) on Google Colab; Jupyter Notebook; SciPy for numerical gradient computation; GNU Octave.

UNIT V: Vector Integration and Integral Theorems

Core 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 in terms of source density; prompting AI to explain connections to Gauss's law and student evaluation of AI explanations.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; SymPy for symbolic integral verification; WolframAlpha free tier; 3D plotting tools.

Course Outcomes (COs)

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 behaviour.	L5 (Evaluate)

CO – PO Articulation Matrix

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

CO	BL	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	L3	3	3	2	2	1	-	-	-	1	2	-	2	1	3
CO2	L4	3	3	3	2	2	-	-	-	1	2	-	3	2	2
CO3	L3	3	2	2	2	1	-	-	-	-	2	-	2	2	3
CO4	L4	3	3	3	3	2	-	-	-	1	2	-	1	3	2
CO5	L5	3	3	2	3	3	-	1	1	2	2	1	1	2	1

CO-PO Weightage

Average weightage based on the COs is shown below; it is aligned to NBA UG POs PO1 to PO11 and kept within the 1–3 correlation scale.

Textbooks

- Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, Wiley, 2018.
- B. S. Grewal, Higher Engineering Mathematics, 44th Edition, Khanna Publishers, 2017.
- G. B. Thomas, M. D. Weir, J. Hass, Thomas' Calculus, 14th Edition, Pearson, 2018.

- R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, 5th Edition, Alpha Science International, 2021.

Reference Books

- Dennis G. Zill and Warren S. Wright, Advanced Engineering Mathematics, 6th Edition, Jones & Bartlett, 2018.
- George B. Arfken, Hans J. Weber, Frank E. Harris, Mathematical Methods for Physicists, 7th Edition, Academic Press, 2012.
- Ian N. Sneddon, Elements of Partial Differential Equations, Dover Publications, 2006.
- H. F. Davis and A. D. Snider, Introduction to Vector Analysis, 7th Edition, Hawkes Publishing, 1995.

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 Sem

Course Code	PHYSICS FOR ELECTRICAL SYSTEMS	L	T	Pr	P	Credits
26AHS00205T	Common to All Branches	3	0	0	0	3

Prerequisites: Engineering Mathematics, Basic Electrical Engineering, Engineering Physics Fundamentals

Course Objectives

- To provide knowledge on wave optics concepts and their applications with basic AI-assisted optical analysis and sensing techniques.
- To introduce the electrical properties of metals and semiconductors with simple AI-assisted conductivity and material analysis methods.
- To equip students with the fundamentals of quantum mechanics and quantum computing with basic AI-assisted secure communication concepts.
- To familiarize students with dielectric and magnetic materials used in power systems along with basic AI-assisted monitoring and predictive analysis techniques.
- To impart knowledge on electromagnetic field theory and its applications in power systems with simple AI-assisted EMI prediction and shielding concepts.

Course Content

UNIT I: Applied Wave Optics (9 hrs)

Core Content: Interference: principle of superposition and coherence, types of interference, interference in thin films (reflection geometry) and Newton's rings, application — 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, application — LiDAR for autonomous sensing in electric vehicles (EV). Polarization: types of polarization, Nicol's prism, half-wave and quarter-wave plates.

AI Integration: AI-powered LiDAR integration for autonomous EV sensing and navigation; Python-based fringe pattern analysis for semiconductor wafer surface profiling.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn / basic DL models.

UNIT II: Electrical Properties of Metals and Semiconductors (8 hrs)

Core Content: Classical free electron theory — merits and demerits; quantum free electron theory — equation for electrical conductivity; Fermi-Dirac statistics; Fermi level and variation with temperature; classification of solids — metals, insulators and semiconductors. 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.

AI Integration: AI-assisted prediction of electrical conductivity in metals and semiconductors; AI-assisted carrier concentration and conductivity analysis in semiconductors.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn.

UNIT III: Introduction to Quantum Mechanics & Quantum Computing (10 hrs)

Core Content: Introduction to quantum mechanics, de-Broglie concept (matter waves), Heisenberg's Uncertainty Principle, wave function, postulates of quantum mechanics, Schrodinger'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 Integration: 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 IV: Dielectrics and Magnetic Materials for Power Systems (9 hrs)

Core Content: Dielectric materials: introduction, basic definitions, types of polarizations, internal fields in solids and the Clausius-Mossotti relation, frequency dependence of the polarisation, dielectric loss; application (qualitative only) — AI-assisted monitoring of solid, liquid and gaseous dielectrics in high voltage applications. Magnetic materials and energy conversion: introduction and classification (dia, para, ferro, ferri, and anti-ferro); hysteresis and eddy current losses in electrical machines; skin effect; soft & hard magnetic materials; applications (basic idea) — transformer cores, armatures, high-frequency inductors and permanent magnets for EV motors and wind generators.

AI Integration: AI-based hysteresis loop analysis and eddy current loss estimation; AI-driven design of permanent magnets for EV motors.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn.

UNIT V: Electromagnetic Fields in Power Systems (9 hrs)

Core Content: Introduction, wave equation in differential form in free space, vector analysis, divergence and curl of electric field and magnetic field (static), Gauss' divergence theorem and Stokes' theorem; derivations of the four Maxwell equations and their significance; Maxwell's equations in vacuum, energy transport and the Poynting vector. Applications (qualitative only): role of Maxwell's equations in electromagnetic interference (EMI), EMI in power converters (switching noise in IGBT/MOSFET-based inverters).

AI Integration: AI-assisted EMI prediction in IGBT/MOSFET-based power converters; AI-optimised electromagnetic shielding for power electronics.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level

CO1	Apply the principles of interference, diffraction, and polarization with basic AI-assisted optical analysis in engineering applications.	L3 (Apply)
CO2	Analyze the electrical properties of metals and semiconductors using basic AI-assisted material characterization and conductivity analysis techniques.	L4 (Analyze)
CO3	Apply the concepts of quantum mechanics and quantum computing with basic AI-assisted secure communication applications.	L3 (Apply)
CO4	Analyse dielectric and magnetic material behaviour in power systems using basic AI-assisted monitoring and predictive maintenance techniques.	L4 (Analyze)
CO5	Apply and analyze Maxwell's equations and electromagnetic field concepts for energy transport, AI-based EMI detection and mitigation in power electronic and smart grid systems.	L3 (Apply) L4 (Analyze)

CO – PO Articulation Matrix

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	3	-	-	-	-	-	-	1	2	2
CO2	3	3	2	2	3	-	-	-	-	-	-	1	3	2
CO3	3	3	2	2	2	-	-	-	-	-	-	1	2	1
CO4	3	3	3	2	3	-	1	-	-	-	-	1	2	2
CO5	3	3	2	3	3	-	-	-	-	-	-	1	1	2

Textbooks

1. Avadhanulu, Kshirsagar & Arun Murthy, Engineering Physics, S. Chand, 2019.
2. D. K. Bhattacharya & Poonam Tandon, Engineering Physics, Oxford Press, 2015.

Reference Books

1. Shatendra Sharma & Jyotsna Sharma, Engineering Physics, Pearson, 2018.
2. S. O. Pillai, Solid State Physics, New Age International.
3. Nielsen & Chuang, Quantum Computation & Quantum Information, Cambridge.
4. Streetman & Banerjee, Solid State Electronic Devices, Pearson.
5. C. Kittel, Introduction to Solid State Physics, Wiley.
6. A. Von Hippel, Dielectrics and Waves, Wiley.

7. D. J. Griffiths, Introduction to Electrodynamics, Cambridge.
8. W. H. Hayt & J. A. Buck, Engineering Electromagnetics, McGraw Hill.

Web Resources & NPTEL Links

- NPTEL – nptel.ac.in
- MIT OpenCourseWare – ocw.mit.edu
- HyperPhysics – hyperphysics.phy-astr.gsu.edu
- Feynman Lectures – feynmanlectures.caltech.edu
- PhET Simulations – phet.colorado.edu
- Library of Congress – loc.gov/rr/scitech/selected-internet/physics.html

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for wave optics and EM field analysis.
- IBM Qiskit (free, open-source) – for quantum computing and quantum cryptography simulations on Google Colab.
- scikit-learn (free, open-source) – for semiconductor property prediction and EMI data analysis.
- OpenEMS (free, open-source) – for 3D electromagnetic field simulation and power system analysis.
- PhET Interactive Simulations (free) – for wave optics, quantum mechanics and semiconductors.

I Year II Sem

Course Code	ELECTROMAGNETIC FIELDS	L	T	Pr	P	Credits
26AEE02201T	I Year II Sem	3	0	0	0	3

Prerequisites: Differential Equations & Vector Calculus

Course Objectives

- To introduce the fundamentals of vector calculus and electrostatics for analyzing electromagnetic field problems.
- To explain the behaviour of conductors and dielectrics in electric fields and analyze capacitance and energy storage mechanisms.
- To study magnetostatic fields, magnetic forces and their applications in electrical engineering systems.
- To understand self and mutual inductance in electromagnetic systems and their applications in electrical devices.
- To analyze time varying electromagnetic fields and Maxwell's equations using modern computational and AI tools.

Course Content

UNIT I: Vector Analysis and Electrostatics

Core Content: **Coordinate Systems:** Rectangular- Cylindrical and Spherical coordinate systems. Differential elements. **Vector Calculus:** Differential length, Area and Volume- Del operator- Gradient of a scalar- Divergence of a vector and Divergence theorem (definition only)- Curl of a vector and Stoke's theorem (definition only)- Laplacian of a scalar. **Electrostatics:** Coulomb's law and Electric field intensity (EFI) – EFI due to Continuous charge distributions (line and surface charge)- Electric flux density- Gauss's law (Maxwell's first equation, $\nabla . D = \rho v$)- Electric Potential- Work done in moving a point charge in an electrostatic field (second Maxwell's equation for static electric fields, $\nabla \times E = 0$)- Potential gradient- Laplace's and Poison's equations.

AI Integration: AI-based vector field visualization. Neural networks for electric field prediction. Machine learning models for electrostatic potential estimation. AI-assisted numerical solution of Laplace and Poisson equations.

Free/Open-Source AI Tools: Python (NumPy, SciPy, SymPy); TensorFlow / PyTorch; Jupyter Notebook; GNU Octave.

UNIT II: Conductors, Dielectrics and Capacitance

Core Content: Electric dipole and dipole moment – Potential and EFI due to an electric dipole- Torque on an Electric dipole placed in an electric field; Behaviour of conductors in an electric field- Polarization- dielectric constant and strength- Current density- conduction and convection current densities- Ohm's law in point form- Continuity equation; Boundary conditions- Capacitance of parallel plate, coaxial capacitors- Energy stored and density in a static electric field.

AI Integration: AI prediction of dielectric properties. Machine learning optimization of capacitor geometry. AI modeling of energy storage behaviour. Data-driven analysis of electric field distribution in dielectric materials.

Free/Open-Source AI Tools: Scikit-Learn; FEniCS (Finite Element simulation); OpenModelica; ParaView.

UNIT III: Magnetic Circuits, Magnetostatics and Magnetic Forces

Core Content: Basic definition of MMF- flux and reluctance- analogy between electrical and magnetic circuits. Biot-Savart's law and its applications viz. Straight current carrying filament, circular, and solenoid current carrying wire – Magnetic flux density and Maxwell's second Equation ($\nabla.B = 0$); Ampere's circuital law and its applications viz. MFI due to an infinite sheet, long filament, solenoid, toroidal current carrying conductor- point form of Ampere's circuital law- Maxwell's third equation ($\nabla \text{ X } H = J$)- Magnetic force, moving charges in a magnetic field – Lorentz force equation- force on a straight and a long current carrying conductor in a magnetic field- force between two straight long and parallel current carrying conductors- Magnetic dipole- Magnetic torque and moment.

AI Integration: Neural networks for magnetic flux density prediction. Machine learning models for magnetic field estimation. Intelligent simulation of magnetic forces.

Free/Open-Source AI Tools: Elmer FEM; OpenEMS; Python (SciPy); PyTorch.

UNIT IV: Self and Mutual Inductance

Core Content: Concept of self and mutual inductance; Dot convention – coefficient of coupling and composite magnetic circuit- analysis of series and parallel magnetic circuits. Determination of self-inductance of a solenoid, toroid, coaxial cable and mutual inductance between a straight long wire and a square loop wire in the same plane – Energy stored and energy density in a magnetic field.

AI Integration: Machine learning prediction of inductance values. Neural networks for magnetic energy storage estimation. Data-driven modeling of mutual inductance.

Free/Open-Source AI Tools: Python (SciPy); FEniCS; OpenEMS; Jupyter Notebook.

UNIT V: Time Varying Fields

Core Content: Faraday's laws of electromagnetic induction- Maxwell's fourth equation ($\nabla \text{ X } E = -\partial B/\partial t$)- integral and point forms of Maxwell's equations- Statically and Dynamically induced EMF- Displacement current- Modification of Maxwell's equations for time varying fields- Poynting theorem and Poynting vector.

AI Integration: Physics-Informed Neural Networks (PINNs) for solving Maxwell equations. Deep learning models for electromagnetic wave propagation. AI-based EM interference prediction. Intelligent analysis of power flow using Poynting vector.

Free/Open-Source AI Tools: DeepXDE (PINNs); Meep electromagnetic simulator; TensorFlow; OpenEMS.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember the concepts of vector calculus, various fundamental laws, self and mutual inductance	BL1 – Remember, BL2 – Understand
CO2	Understand the concepts of electrostatics, conductors, dielectrics, capacitance, magneto statics, magnetic fields, time varying fields, self and mutual inductances	BL3 – Apply
CO3	Apply vector calculus, Coulomb's law, Gauss's law, Ohm's law, Biot-Savart's law, Ampere's circuital law, Maxwell's third equation, self and mutual inductances, Faraday's laws, Maxwell's	BL4 – Analyze

CO	Course Outcome Statement	Bloom's Level
	fourth equation.	
CO4	Analyze vector calculus, electrostatic fields, and conductor behavior, and apply AI and machine learning techniques for prediction, simulation, and analysis in electrostatic systems.	BL3 – Apply, BL4 – Analyze
CO5	Analyze magnetic force, moving charges in magnetic fields, self and mutual inductance of cables and wires, and time-varying fields, and apply AI techniques for field prediction, inductance estimation, and electromagnetic system analysis.	BL4 – Analyze, Design Concept

CO – PO Articulation Matrix

Note: 1 – Low, 2 – Medium, 3 – High, '-' – No Correlation

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	BL1, BL2	3	3	3	2	1	-	-	-	-	-	1	3	3	3
CO2	BL3	3	3	3	2	1	-	-	-	-	-	1	3	3	3
CO3	BL4	3	3	3	2	1	-	-	-	-	-	1	3	3	3
CO4	BL3, BL4	3	3	3	2	1	-	-	-	-	-	1	3	3	3
CO5	BL4	3	3	3	2	1	-	-	-	-	-	1	3	3	3

Textbooks

1. Elements of Electromagnetics by Matthew N O Sadiku, Oxford Publications, 7th Edition, 2018.
2. Engineering Electromagnetics by William H. Hayt & John A. Buck, Mc. Graw-Hill, 7th Edition, 2006.

Reference Books

1. Introduction to Electro Dynamics by D J Griffiths, Prentice-Hall of India Pvt. Ltd, 2nd Edition.
2. Electromagnetic Field Theory by K.A. Gangadhar and P.M. Ramanathan, Khanna Publishers, 16th Edition, 2015.
3. Electromagnetic Field Theory by Yaduvir Singh, Pearson India, 1st Edition, 2011.
4. Fundamentals of Engineering Electromagnetics by Sunil Bhooshan, Oxford University Press, 2012.

5. Schaum's Outline of Electromagnetics by Joseph A. Edminister, Mahamood Navi, 4th Edition, 2014.

Web Resources & NPTEL Links

- NPTEL: Electromagnetic Field Theory: https://nptel.ac.in/courses/108106073
- MIT OpenCourseWare – Electromagnetics and Applications: https://ocw.mit.edu/courses/6-013-electromagnetics-and-applications/
- MIT Computational Electromagnetics: https://ocw.mit.edu/courses/computational-electromagnetics/
- Physics Informed Neural Networks: https://maziarraissi.github.io/PINNs/
- Python Scientific Computing: https://scipy.org

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, SymPy) – free on Jupyter Notebook / Google Colab for vector field and electrostatic computation
- TensorFlow / PyTorch (free, open-source) – for AI/ML based field prediction and PINN modelling
- Scikit-Learn (free, open-source) – for regression and optimization of dielectric and capacitor properties
- FEniCS (free, open-source) – FEM-based electrostatic and magnetostatic simulation
- OpenModelica (free, open-source) – for system-level electromagnetic modelling
- ParaView (free, open-source) – for visualization of field simulation results
- Elmer FEM (free, open-source) – magnetostatic and electromagnetic finite element analysis
- OpenEMS (free, open-source) – electromagnetic field simulation with Python interface
- DeepXDE (free, open-source) – for solving Maxwell's equations using PINNs
- Meep electromagnetic simulator (free, open-source) – for time-varying field and wave propagation studies
- GNU Octave (free, open-source) – for numerical vector calculus computation

I Year II Sem

Course Code	ENGINEERING GRAPHICS	L	T	Pr	P	Credits
26AME03201T	Common to All Branches	1	0	4	0	3

Prerequisites: Basic Geometry, Fundamental Python Programming

Course Objectives

- Familiarise with the BIS conventions, geometric construction standards, and engineering scales.
- Develop visualization skills to interpret and project points, lines, and regular planes into 2D drawing sheets.
- Impart foundational knowledge on projecting regular 3D solids and reading directional geometric orientations.
- Enable the spatial reconstruction of sliced solids and flat unfolded developments of various structural shells.
- Introduce CAD tools combined with generative AI software to construct multi-view isometric representations from 3D models.

Course Content

UNIT I: Introduction to Engineering Graphics, Curves, and Scales

Unit Objectives:

- To instruct students on basic scaling parameters.
- To familiarize with continuous geometric drafting styles.

Unit Outcome:

- Understand and master standard curve outlines and be able to configure precise dimensional layout scales.
- Draw accurate engineering curves (ellipses, parabolas, and hyperbola profiles) by following standard methods.

Core Content: Introduction: Principles of Engineering Graphics, signboards, BIS conventions, and lettering. Curves: Construction of Polygons. Conic sections: Ellipse, Parabola, and Hyperbola using eccentricity/general methods. Cycloidal curves and Involutes. Scales: Plain, Diagonal, and Vernier scales.

Hands-on Experience: Manual draft generation of an ellipse and diagonal scale, paired with a software verification loop.

AI Integration: AI prompt engineering for generating geometric profiles; algorithmic generation of conic curves using mathematical code blocks; parsing hand-drawn geometric sketches into clean vector shapes using Computer Vision (CV) edge detection (Canny Edge/Hough Transform).

UNIT II: Orthographic Projections of Points, Lines and Planes

Unit Objectives:

- To visualize and construct orthographic projections of points and complex lines in multi-planar orientations.
- To apply geometric transform matrices to determine physical properties (true length, inclination) programmatically.

Unit Outcomes:

- Able to construct projections of lines positioned at complex angles relative to the HP and VP reference planes.
- Able to develop matrix calculation scripts that instantly map 2D line projections back to their 3D space orientations.

Core Content: Principles of orthographic projections, projection of points in all four quadrants. Projection of straight lines inclined to one or both reference planes, traces of lines, determination of true lengths and true inclinations. Projections of Planes: regular planes perpendicular to both reference planes, parallel to one reference plane and inclined to the other reference plane; plane inclined to both the reference planes.

Hands-On Experience & Application: Project: Construct a transformation script where a user inputting 2D projection endpoints automatically outputs the 3D line representation, true spatial length, and inclination angles without manual geometric tracking. Application: Calibration setups for stereoscopic robotic arms tracking trajectories across multiple camera planes.

AI Integration: Machine learning pipelines for 3D coordinate estimation from 2D views; multi-view structural tracking; training neural models to predict true line length and orientation angles based solely on coordinate projections.

UNIT III: Projection of Regular Solids

Unit Objectives:

- To draft multi-view configurations of flat planes and three-dimensional solid bodies in skewed spatial attitudes.
- To employ deep learning classifiers to identify, sort, and tag complex geometric features on mechanical components.

Unit Outcomes:

- Generate multi-stage projection views for complex prisms, pyramids, and curved solids.
- Build lightweight feature-detection pipelines that recognize primitive solid features from standard engineering layouts.

Core Content: Projection of Regular Solids: Types of solids: Polyhedra and Solids of revolution. Projections of solids in simple positions: Axis perpendicular to horizontal plane, Axis perpendicular to vertical plane and Axis parallel to both the reference planes, Projection of Solids with axis inclined to one reference plane and parallel to another plane.

Hands-On Experience & Application: Project: 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 an orthographic engineering layout image. Application: Automated component sorting inside factory staging bays and conveyor streams.

AI Integration: Object recognition models (e.g., YOLO / MobileNet) trained to identify standard solid geometries; automated multiview projection generation from 3D CAD meshes using programmatic render layers.

UNIT IV: Sections of Solids, Development of Surfaces, and AI Generative Design

Unit Objectives:

- To section basic regular solids accurately and project the actual structural profile of the cut face.
- To develop unfolded 3D structures into highly accurate flat-pattern sheet configurations optimized for assembly floor environments.

Unit Outcomes:

- Able to draw exact sectional views and map out flat spatial developments for complex intersected structures.
- Set up structural optimization pipelines that reduce industrial sheet nesting material waste.

Core Content: Sections of Solids: Perpendicular and inclined section planes, sectional views and true shape of section, sections of solids in simple position only. Development of Surfaces: Methods of Development: Parallel line development and radial line development. Development of a cube, prism, cylinder, pyramid and cone.

Hands-On Experience & Application: Project: Use a nesting optimization script (e.g., using genetic algorithms or standard heuristics) to arrange several custom flat-surface solid developments onto a finite raw sheet grid, actively minimizing material margins and cut lines. Application: Sheet metal stamping pattern automation in automotive and aerospace chassis manufacturing.

AI Integration: Generative Adversarial Networks (GANs) for generating flat-pattern configurations; algorithmic optimization of surface layouts to minimize sheet manufacturing scrap; predictive toolpath generation using physics-guided heuristic networks.

UNIT V: Isometric Projections, Conversions, and 2D-to-3D Deep Reconstruction

Unit Objectives:

- To construct proportional isometric views using isometric scales for compound physical structures.
- To convert fluidly between 2D orthographic perspectives and comprehensive 3D isometric designs.

Unit Outcomes:

- Able to produce detailed isometric drawings from multi-view orthographic sheets and vice-versa.
- Able to transform raw 2D technical drawings into interactive 3D model formats.

Core Content: Principles of isometric projection, isometric scale, isometric views/projections of planes, simple and compound solids. Conversion of isometric views to orthographic views and vice-versa. Computer graphics: Creating 2D & 3D drawings of objects including PCB and Transformations using AutoCAD (Not for end examination).

Hands-On Experience & Application: Project: Use a pretrained depth estimation or pixel-to-voxel neural model to parse a set of front, top, and side drawings, compiling them directly into an interactive 3D point cloud or boundary surface rendering. Application: Instantly generating 3D printable mechanical files directly from archival 2D layout drafting logs.

AI Integration: 2D-to-3D depth reconstruction using Neural Radiance Fields (NeRF) or specialized generative models; auto-generation of 3D CAD models from simple multi-view engineering layout sketches.

Course Outcomes (COs)

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.	L3 (Apply)
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.	L6 (Create)

CO – PO Articulation Matrix

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	-	-	-	-	-	2	-	1	2	1
CO2	3	3	2	-	-	-	-	-	-	2	-	1	3	2
CO3	3	3	2	1	-	-	-	-	-	2	-	1	2	3
CO4	3	3	3	1	2	-	-	-	-	2	-	2	3	2
CO5	3	2	2	-	3	-	-	-	1	3	-	2	2	1

Textbooks

1. Engineering Drawing by N.D. Bhatt, Charotar Publishing House, 2016.
2. Engineering Drawing with an Introduction to AutoCAD, Dhananjay Jolhe, Tata McGraw Hill, 2017.

Reference Books

3. Engineering Graphics and Design by Pradeep Jain, A.P. Gautam, Abhishek Bhargava, Khanna Publishing.
4. Engineering Drawing, K.L. Narayana and P. Kannaiah, Tata McGraw Hill, 2013.
5. Engineering Drawing, M.B. Shah and B.C. Rana, Pearson Education Inc, 2009.

Web Resources & NPTEL Links

- Classical Drafting Fundamentals: NPTEL Engineering Graphical Design Course by IIT Kharagpur – https://nptel.ac.in
- Computer Vision and Processing: OpenCV Official Python Bootcamps and Tutorials – https://opencv.org
- Generative Design and Modeling Insights: MIT 6.S192 Deep Generative Modeling Platforms – https://ocw.mit.edu

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SymPy) on Google Colab (free) – for algorithmic generation of conic curves and geometric profiles
- OpenCV (free, open-source) – Computer Vision edge detection (Canny Edge/Hough Transform) for parsing hand-drawn sketches
- scikit-learn (free, open-source) – classification notebooks for identifying solid geometries
- YOLO / MobileNet (open-source pretrained models) – object recognition of standard solid geometries
- FreeCAD / AutoCAD (student version) – CAD modelling, 3D meshes and multiview projection generation
- Google Colab (free) – for GAN-based flat-pattern generation and nesting optimization scripts

I Year II Sem

Course Code	ELECTRICAL ENGINEERING WORKSHOP PRACTICES	L	T	Pr	P	Credits
26AEE02206P	I Year II Sem	0	0	0	2	1

Course Objectives

- To know about different tools, abbreviations and symbols in Electrical Engineering.
- To learn about types of measuring instruments to measure electrical quantities.
- To gain knowledge on different types of earthing and earth resistance.
- To study different types of wiring.
- To study color code, resistors, ICs, transistors, capacitors, diodes, SCRs, IGBTs.

List of Practices

Any 12 of the following practices are to be performed:

1. Study of Introduction to Electrical tools, symbols and abbreviations.
2. Study of types of sizes of wires and making "T" joint and straight joint for wires.
3. Measurements of Electrical quantities (like Voltage, Current, Power, Power factor in RLC circuits).
4. Measurements of Energy (using Single phase and Three phase Energy meter) by connecting different loads.
5. Study of earthing and measurement of earth resistance.
6. Study and performance of residential wiring (using Energy meter, Fuses, Switches, Indicator, Lamps, etc.).
7. Study of Fluorescent lamp wiring.
8. Study of various electrical gadgets (CFL and LED).
9. Study of PV Cell.
10. Study of cross-sectional aspects Induction motor and Transformer.
11. Assembly of choke or small transformer.
12. Study of trouble shooting of electrical equipment's (fan, iron box, mixer-grinder, etc.).
13. Introduction to basics of electronic components: Solder practice, Multi meter, Power supply.
14. Measurement of wire gauges using gauge meter.
15. Identification of color code, resistors, ICs, Transistors, capacitors, diodes, SCRs, IGBTs.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the functions, symbols, abbreviations, safety practices, electrical tools, wiring accessories, electronic components, and basic electrical/electronic equipment used in laboratory and residential applications.	BL1 – Remember, BL2 – Understand

CO	Course Outcome Statement	Bloom's Level
CO2	Apply electrical wiring techniques, wire joint preparation, soldering practices, earthing methods, and measurement procedures for voltage, current, power, power factor, energy, and earth resistance using standard instruments.	BL3 – Apply
CO3	Analyze the operating principles and performance characteristics of RLC circuits, fluorescent lamps, PV cells, transformers, induction motors, energy meters, and electrical gadgets such as CFLs and LEDs.	BL4 – Analyze
CO4	Apply and analyze troubleshooting techniques to identify faults and evaluate the performance of electrical and electronic equipment including fans, iron boxes, mixer-grinders, transformers and basic electronic circuits.	BL3 – Apply, BL4 – Analyze
CO5	Design simple residential wiring systems, basic transformer/choke assemblies, and elementary electronic circuits by selecting suitable wires, components, protective devices, and measurement instruments for safe and efficient operation.	BL4 – Analyze, Design Concept

CO – PO Articulation Matrix

Note: 1 – Low, 2 – Medium, 3 – High, '-' – No Correlation

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	BL1, BL2	3	-	-	-	1	-	2	3	-	-	-	2	1	1
CO2	BL3	3	3	-	-	1	-	1	1	-	-	-	2	2	1
CO3	BL4	2	-	3	-	-	-	2	2	-	-	-	2	2	1
CO4	BL3, BL4	3	2	-	3	-	-	1	3	-	-	-	2	2	1

Reference Books

1. Lab manual of Electrical Engineering by TTTI, Chennai.

I Year II Sem

Course Code	PHYSICS FOR ELECTRICAL SYSTEMS LAB	L	T	Pr	P	Credits
26AHS00205P		0	0	0	3	1.5

Prerequisites: Engineering Physics (Theory)

Course Objectives

- Provide hands-on experimental verification of the physical principles underlying optics, electromagnetism, and semiconductor behaviour.
- Develop skill in the use of standard laboratory instruments for measuring optical, electrical, and magnetic quantities.
- Train students to record, analyse, and interpret experimental data with appropriate error estimation.
- Introduce AI/ML and Python-based computational tools for simulation and analysis of physics experiments.

Course Content – List of Experiments

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 of a LASER using a diffraction grating.
6. Estimation of Planck's constant using the photoelectric effect.
7. Determination of e/m of an electron by J.J. Thomson method.
8. Determination of the Curie temperature and dielectric constant of a ferroelectric material.
9. Study the variation of B versus H by magnetizing the magnetic material (B-H curve).
10. Determination of the resistivity of a semiconductor by the four-probe method.
11. Determination of the energy gap of a semiconductor using a p-n junction diode.
12. Determination of Hall voltage and Hall coefficient of a given semiconductor using the Hall effect.
13. Verification of the inverse square law for a magnetic field using a Hall probe.
14. Measurement of self-inductance of a coil using LCR meter and resonance method.
15. Determination of magnetic field along the axis of a current-carrying circular coil by Stewart & Gee's Method.

AI Integration

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

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Explain the physical principles underlying the optical, electromagnetic, and semiconductor experiments performed in the laboratory.	L2 (Understand)
CO2	Conduct experiments to determine optical constants such as dispersive power, wavelength, and radius of curvature.	L3 (Apply)
CO3	Measure electrical and magnetic properties of materials including resistivity, energy gap, and Hall coefficient.	L3 (Apply)
CO4	Analyse experimental data, estimate errors, and validate results using Python-based computational tools.	L4 (Analyze)
CO5	Evaluate the agreement between experimental and theoretical values using AI/ML-assisted data analysis.	L5 (Evaluate)

CO – PO Articulation Matrix

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1	-	-	-	-	-	-	-	-	-	1	3	1
CO2	3	2	-	2	1	-	-	-	1	-	-	1	2	3
CO3	3	2	-	2	1	-	-	-	1	-	-	1	3	2
CO4	2	3	-	3	2	-	-	-	1	1	-	2	2	2
CO5	2	3	-	2	2	-	-	-	1	1	-	2	2	1

Textbooks

As prescribed by the department – refer to the corresponding Engineering Physics theory course textbooks and the laboratory manual issued by the Department of Physics.

Reference Books

As prescribed by the department – refer to the corresponding Engineering Physics theory course reference books and the laboratory manual issued by the Department of Physics.

Web Resources & NPTEL Links

- HyperPhysics – hyperphysics.phy-astr.gsu.edu
- PhET Simulations – phet.colorado.edu
- IIT Bombay Virtual Labs (Physical Sciences) – https://iitb.vlabs.co.in/discipline.html?discipline=Physical_Sciences

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for wave optics and EM field analysis
- scikit-learn (free, open-source) – for semiconductor property prediction and EMI data analysis
- OpenEMS (free, open-source) – for 3D electromagnetic field simulation and power system analysis

I Year II Sem

Course Code	COMMUNICATIVE ENGLISH LAB	L	T	Pr	P	Credits
26AHS00201P	Common to All Branches	0	0	0	2	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.

Course Content – 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 feedback via ELSA Speak	ELSA Speak, Google Speech Recognition, Speechling	CO1, CO5
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	AI generates arguments on both sides; evaluate logical structure	ChatGPT, Claude.ai, Kialo Edu	CO2, CO3

		fallacy identification	using ChatGPT		
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

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
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

CO – PO / PSO Articulation Matrix

Correlation Level: 3 = High | 2 = Medium | 1 = Low | – = No Correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	-	-	2	-	1	2	-	-	-	-	3	2	2
CO2	2	-	-	2	-	2	1	-	-	-	-	2	3	3
CO3	2	-	-	3	-	3	2	-	-	-	-	2	3	3
CO4	2	-	-	2	-	2	1	-	-	-	-	3	2	2
CO5	1	-	-	2	3	2	2	-	-	-	-	2	3	2

Mapping Legend: PO1 – Grammar, vocabulary & technical language (2); PO4 – GD, presentations, professional writing (3); PO5 – AI tools for language learning (3); PO6 – Workplace etiquette & netiquette (2); PO7 – Self-paced AI-assisted improvement (2).

Textbooks

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.

Reference Books

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.

Web Resources & NPTEL Links

- 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)

- YouTube – English with Lucy (@EnglishwithLucy): Vocabulary, speaking, British accent
- YouTube – Learn English with TV Series (@LearnEnglishWithTVSeries)
- YouTube – Rachel's English (@rachelsenglish): American accent and pronunciation
- YouTube – TED-Ed (@TEDEd): Critical thinking and communication ideas
- YouTube – Spoken English Hub (@SpokenEnglishHub): Indian context, interview prep
- YouTube – JenniferESL (@JenniferESL): Grammar and business English
- NPTEL / SWAYAM – Developing Soft Skills and Personality, IIT Kanpur – https://swayam.gov.in
- Coursera (Free Audit) – Business English Communication Suite, University of Washington – https://coursera.org
- Coursera – Speak English Professionally: In Person, Online & On the Phone, Georgia Tech – https://coursera.org
- NPTEL – English and Academic Preparation, IIT Bombay – https://nptel.ac.in
- NPTEL / SWAYAM – Technical English for Engineers, IIT Roorkee – https://swayam.gov.in

Recommended Free & Open-Source AI/Computational Tools

- 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 (free, open-source) – Audio Recording Software: record and review pronunciation exercises
- Google Meet / Zoom – Video Conferencing: mock interviews and virtual presentations

Canva / Beautiful.ai – AI Pr

I Year II Sem

Course Code	HEALTH & WELLNESS, YOGA AND SPORTS	L	T	Pr	P	Credits
26AHS00201L	Common to All Branches	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 Content

UNIT I: Health, Wellness and Preventive Healthcare

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: ULO1.1 Define health, wellness and fitness and describe the seven 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 II: Yoga, Pranayama, Meditation and Mind-Body Practice

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: 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 III: Physical Fitness, Sports Skills and Sports Technology

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: 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].

AI Integration – 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.

Course Outcomes (COs)

CO	Course Outcome Statement	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)

CO – PO / PSO Articulation Matrix

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	-	-	-	-	3	2	-	-	-	-	-	-	-
CO2	-	-	3	-	-	3	3	-	-	-	-	-	-	-
CO3	-	2	3	3	-	3	3	-	-	-	-	2	-	-
CO4	-	-	3	-	-	2	2	-	3	-	-	-	-	-
CO5	-	3	-	3	2	3	2	-	-	-	-	2	-	-

Textbooks

16. Mangal, S. K. & Mangal, Shubhra. Health and Physical Education. PHI Learning Pvt. Ltd., 2021.
17. Broad, William J. The Science of Yoga. Simon & Schuster, 2012.

Reference Books

18. Silva, Mira & Mehta, Shyam. Yoga: The Iyengar Way. Dorling Kindersley, 2019.
19. Clover, Jim. Sports Medicine Essentials. Cengage Learning, 2020.
20. Bucher, Charles A. Foundations of Physical Education & Sport. McGraw-Hill Education, 2018.
21. Williams, Mark & Penman, Danny. Mindfulness: An Eight-Week Plan for Finding Peace. Rodale Books, 2011.

Web Resources & NPTEL Links

- 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: Surya Namaskar – Complete 12-Step Tutorial
- YouTube: Pranayama for Beginners (Swami Ramdev)
- YouTube: Mindfulness Meditation (10 Min)
- YouTube: Balanced Nutrition Explained (ICMR)
- YouTube: Sports Injury Prevention Tips
- YouTube: Mental Health Awareness for Students
- YouTube: Yoga for Stress Relief – Isha Foundation
- YouTube: Fit India Movement – PM Modi Launch

Recommended Free & Open-Source AI/Computational Tools

- Google Fit / Apple Health (free) – BMI, step count and sleep tracking with AI health-insight reports
- NutriAI / Eat Right Now (free tier) – AI-based personalised diet planning
- YogaNotch / KAIA Health (free tier) – ML-based yoga pose correction using camera feedback
- Woebot / Wysa (free tier) – AI mental-health chatbot for reflective wellness practice

Python (Pandas, Matplotlib

I Year II Sem

Course Code	CYBERSECURITY AWARENESS FOR ENGINEERS	L	T	Pr	P	Credits
26ACS05101M	Common to All Branches	2	0	0	0	0

Course Objectives

The course aims to:

- Introduce students to the fundamentals of cybersecurity 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 cybersecurity concerns in cloud, digital platforms, and AI-enabled environments.

Course Content

UNIT–I: Introduction to Cyberspace and Cybersecurity

Core Content: Introduction to cyberspace and cybersecurity – Evolution of information systems and Internet – Importance of cybersecurity 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: Awareness-level discussion of how AI-driven monitoring tools support digital identity and threat-actor profiling; use of AI chatbots (ChatGPT/Claude) to explain CIA Triad concepts and security governance basics through interactive Q&A.

UNIT–II: Cyber Threats, Cybercrime and Social Engineering

Core Content: Cyber threats and vulnerabilities – Cybercrime and cyber offenders – Malware: Virus, Worm, Trojan, Spyware, Ransomware – Cyberattack 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 deepfakes – Case studies of major cyber incidents.

AI Integration: AI-enabled cyber threats and deepfakes are studied as an emerging attack category; awareness exercises using deepfake-detection demo tools and AI-based phishing-email classifiers to illustrate how AI is used both offensively and defensively.

UNIT–III: Personal Cyber Safety and Cyber Security Technologies

Core 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: Safe use of AI tools and digital platforms is studied explicitly, including data-privacy considerations when using generative AI tools, and AI-based password-strength and breach-checking utilities.

UNIT–IV: Privacy, Cyber Ethics and Cyber Laws

Core 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 – Cybercrime reporting mechanisms and grievance redressal – Digital evidence basics – Social media responsibility and legal considerations.

AI Integration: Discussion of ethical and legal implications of AI-generated content, data-privacy regulations relevant to AI systems, and the use of AI-assisted tools for drafting responsible-use policies and grievance reports.

UNIT–V: Cyber Hygiene and Secure Digital Practices

Core Content: Cybersecurity 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 – Cybersecurity awareness in IoT and emerging technologies – Incident awareness and cyber hygiene best practices.

AI Integration: Secure use of social media and AI tools is studied as a standing cyber-hygiene practice; case-based awareness of AI-enabled IoT threats and use of AI-based security-awareness training simulators.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Explain fundamental cybersecurity concepts and their importance in digital environments.	L2 (Understand)
CO2	Identify common cyber threats, cyberattacks, 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.	L3 (Apply)
CO5	Develop cyber hygiene habits and respond appropriately to common cybersecurity incidents.	L3/L6 (Apply/Create)

CO – PO Articulation Matrix

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	1	-	-	1	2	-	1	-	-	-	1	1	1
CO2	2	2	-	1	1	2	-	1	-	-	-	1	1	1
CO3	2	2	1	1	2	2	-	2	1	1	-	2	1	1
CO4	1	1	-	-	1	3	-	3	1	1	-	2	1	1
CO5	1	2	1	1	2	2	-	2	1	1	-	2	1	1

Textbooks

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

Reference Books

3. Brooks, Charles J., Grow, Christopher, Craig, Philip & Short, Donald. Cybersecurity Essentials. Wiley.
4. Stallings, William & Brown, Lawrie. Computer Security: Principles and Practice. Pearson.
5. Whitman, Michael E. & Mattord, Herbert J. Principles of Information Security. Cengage Learning.
6. Kizza, Joseph Migga. Guide to Computer Network Security. Springer.
7. Cole, Eric. Cybersecurity for Beginners. Wiley.

Web Resources & NPTEL Links

- National Cyber Security Awareness Portal (India)
- CERT-In (Indian Computer Emergency Response Team)
- National Cyber Crime Reporting Portal
- NIST Cybersecurity Framework Resources
- OWASP Foundation
- Cisco Networking Academy – Introduction to Cybersecurity

Recommended Free & Open-Source AI/Computational Tools

- Have I Been Pwned (free) – breach-checking utility for personal account awareness
- Google Password Checkup (free) – AI-assisted password strength and breach detection
- VirusTotal (free) – AI/ML-assisted malware and URL scanning
- Microsoft Defender (free tier) – AI-based threat detection awareness demonstrations

- ChatGPT / Claude – for cyber-scenario simulation, phishing-awareness quizzes and case-study discussion
- Cisco Networking Academy modules (free) – for structured cybersecurity awareness training

II Year I Sem II-Year I-Sem

Course Code	COMPLEX VARIABLES AND PROBABILITY THEORY	L	T	Pr	P	Credits
26AHS00301T	Common to CE, ME, ECE & EEE Branches	3	0	0	0	3

Course Objectives

- Develop a thorough understanding of complex variable theory, analyticity, Cauchy–Riemann conditions, and harmonic functions as foundations for engineering analysis.
- Apply complex integration techniques, series expansions, and the residue theorem to evaluate contour integrals and real definite integrals arising in engineering problems.
- Formulate probabilistic models using axioms, random variable theory, and standard distributions, and compute correlation and regression measures for data analysis.
- Integrate analytical methods with AI-assisted computational tools for visualization, simulation, and verification of complex and probabilistic computations.
- Develop outcome-oriented problem-solving skills aligned with NEP 2020 and modern engineering applications.

Course Content

UNIT I: Complex Functions and Analyticity

Core Content: Functions of a complex variable; limits and continuity; differentiability; analyticity; Cauchy–Riemann equations in Cartesian coordinates; Cauchy–Riemann equations in polar coordinates; harmonic functions; conjugate harmonic functions; Milne–Thomson method.

AI Integration: Python-based visualization of complex functions and their mappings; symbolic verification of Cauchy–Riemann equations using SymPy; AI-assisted plotting of harmonic and conjugate harmonic surfaces; use of ChatGPT/Copilot to explain analyticity and the Milne–Thomson construction.

Free/Open-Source AI Tools: Python (NumPy, Matplotlib, SymPy) on Google Colab; Jupyter Notebook; WolframAlpha free tier for verification; GNU Octave.

UNIT II: Complex Integration

Core Content: Line integrals in the complex plane; Cauchy’s integral theorem; Cauchy’s integral formula; Generalized Cauchy’s integral formula; Liouville’s theorem; Maximum modulus theorem.

AI Integration: Numerical evaluation of complex line integrals using SciPy; AI-assisted verification of Cauchy’s integral theorem results; visualization of contour paths in the complex plane using Matplotlib; use of AI tools to interpret Liouville’s theorem and maximum modulus principle.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib, SymPy) on Google Colab; Jupyter Notebook; WolframAlpha free tier; GNU Octave.

UNIT III: Series Expansions and Residues

Core Content: Taylor's series; Laurent's series; zeros and singularities: isolated, removable, poles, essential; residues; Cauchy's residue theorem; evaluation of real definite integrals using the residue theorem.

AI Integration: SymPy-based Taylor and Laurent series generation for complex functions; AI-assisted classification of singularities and residue computation; Python visualization of convergence regions; numerical validation of definite integrals using SciPy and comparison with residue-based results.

Free/Open-Source AI Tools: Python (SymPy, NumPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; WolframAlpha free tier for symbolic verification.

UNIT IV: Probability and Random Variables

Core Content: Sample space and events; axioms of probability; addition and multiplication theorems; conditional probability; Bayes' theorem; random variables: discrete and continuous; probability mass function; probability density function; cumulative distribution function; mathematical expectation and variance.

AI Integration: Python-based simulation of sample spaces and probability experiments; visualization of PMF, PDF, and CDF using Matplotlib; AI-assisted explanation of Bayes' theorem with real-world examples; NumPy/SciPy computation of expectation and variance for standard distributions.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib) on Google Colab; SymPy for symbolic probability computations; Jupyter Notebook; scikit-learn for exploratory data modelling.

UNIT V: Probability Distributions

Core Content: Binomial distribution: properties, mean and variance; Poisson distribution: properties, Poisson approximation to binomial; normal distribution: properties, standard normal curve, areas under normal curve; correlation coefficient; rank correlation; regression lines and regression coefficients.

AI Integration: SciPy-based computation and visualization of binomial, Poisson, and normal distributions; AI-assisted plotting of standard normal curve and computation of area probabilities; Python-based correlation and regression analysis on engineering data sets; use of AI tools to interpret regression coefficients and correlation strength.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib, pandas) on Google Colab; scikit-learn for regression and correlation analysis; Jupyter Notebook; WolframAlpha free tier; GNU Octave.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Recall definitions of limits, continuity, and differentiability of complex functions and state the Cauchy–Riemann equations in Cartesian and polar forms.	L1 (Remember)

CO2	Apply Cauchy's integral theorem, Cauchy's integral formula, Liouville's theorem, and the maximum modulus theorem to evaluate complex line integrals.	L3 (Apply)
CO3	Analyse complex functions using Taylor's and Laurent's series, classify singularities, and evaluate real definite integrals by the residue theorem.	L4 (Analyse)
CO4	Understand the axioms and theorems of probability and explain the properties of discrete and continuous random variables including expectation and variance.	L2 (Understand)
CO5	Evaluate probabilities using binomial, Poisson, and normal distributions and interpret correlation and regression relationships in engineering data sets.	L5 (Evaluate)

CO – PO Articulation Matrix

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

CO	BL	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	L1	3	1	-	-	-	-	-	-	-	-	-	3	2	1
CO2	L3	3	2	1	-	-	-	-	-	-	-	-	3	2	1
CO3	L4	3	3	2	2	1	-	-	-	-	-	-	3	2	1
CO4	L2	3	2	-	1	-	-	-	-	-	-	-	3	2	1
CO5	L5	3	3	1	2	1	1	-	-	1	1	-	3	2	1

CO-PO Weightage

Average weightage based on the COs is shown below; it is aligned to NBA UG POs PO1 to PO11 and kept within the 1–3 correlation scale.

Textbooks

1. Kreyszig, E. Advanced Engineering Mathematics, 10th Edition, Wiley, 2011.
2. Walpole, R. E., Myers, R. H., Myers, S. L., and Ye, K. Probability & Statistics for Engineers and Scientists, 9th Edition, Pearson, 2016.
3. Churchill, R. V. and Brown, J. W. Complex Variables and Applications, 9th Edition, McGraw-Hill, 2013.
4. Miller, J. and Freund, J. E. Probability and Statistics for Engineers, 9th Edition, Pearson, 2014.

5. Jain, R. K. and Iyengar, S. R. K. Advanced Engineering Mathematics, 5th Edition, Alpha Science International, 2021.

Reference Books

1. Ross, S. M. Introduction to Probability and Statistics for Engineers and Scientists, 5th Edition, Academic Press, 2014.
2. Grewal, B. S. Higher Engineering Mathematics, 44th Edition, Khanna Publishers, 2021.
3. Zill, Dennis G. A First Course in Complex Analysis with Applications, 3rd Edition, Jones & Bartlett, 2018.
4. Gupta, S. C. and Kapoor, V. K. Fundamentals of Mathematical Statistics, 12th Edition, Sultan Chand & Sons, 2020.

Web Resources & NPTEL Links

- NPTEL: Complex Analysis – https://nptel.ac.in
- NPTEL: Probability and Statistics – https://nptel.ac.in
- MIT OpenCourseWare – Complex Variables with Applications: https://ocw.mit.edu
- MIT OpenCourseWare – Probability and Statistics: https://ocw.mit.edu
- Python SymPy Documentation: https://docs.sympy.org
- Google Colab for Python: https://colab.research.google.com
- SciPy Documentation: https://docs.scipy.org
- NumPy Documentation: https://numpy.org/doc

Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, Matplotlib) – FREE on Google Colab for complex function visualization, numerical integration, and probability computations.
- SymPy (free, open-source) – For symbolic complex analysis, series expansions, residue computation, and probability expressions.
- scikit-learn (free, open-source) – For regression analysis, correlation computation, and exploratory data modelling.
- Jupyter Notebook (free, open-source) – For interactive computation and lab documentation.
- pandas (free, open-source) – For data manipulation and statistical analysis in probability and regression tasks.
- GNU Octave (free, open-source) – MATLAB-like numerical computing for matrix and complex analysis workflows.
- WolframAlpha (free tier) – For on-demand symbolic verification of integrals, series, and probability computations.

II Year I-Sem

Course Code	UNIVERSAL HUMAN VALUES – UNDERSTANDING HARMONY AND ETHICAL HUMAN CONDUCT	L	T	Pr	P	Credits
26AHS00302T	Common to All Branches	3	0	0	0	3

Course Overview

This course, as recommended by AICTE and aligned with NBA guidelines, explores the foundational principles of human existence, values, and ethical conduct through a systematic, self-exploratory approach. It draws from Indian philosophical traditions, particularly the works of A. Nagraj, and integrates contemporary perspectives including AI ethics. Students are encouraged to critically examine prevailing beliefs, use AI as a reflective tool, and develop a coherent value framework for their professional and personal lives.

Course Objectives

- Introduce the need for value education and help students distinguish natural acceptance from socially imposed beliefs.
- Develop the ability to achieve harmony between the Self and the Body, and among the four dimensions of human being.
- Enable students to extend harmony from the individual to the family and society through trust and respect.
- Cultivate an understanding of harmony in nature and existence, including ecological responsibility and sustainability.
- Foster ethical human conduct and professional ethics, including a critical understanding of AI ethics and bias.

Course Content

UNIT 1: Introduction to Value Education and Human Consciousness

Core Content: Need for Value Education: Crisis of values in modern society. Human Being: A co-existence of Self (Jeevan) and Body; Consciousness as a distinguishing feature. Natural Acceptance: The innate human capacity to know what is right. Activities of the Self: Knowing, Assuming, Recognising, Fulfiling. Difference between Animal Consciousness and Human Consciousness.

AI Integration: AI Tool: ChatGPT – present ethical dilemmas to AI and compare AI responses with naturally accepted human values. Reflection Activity: Use an AI journalling assistant (Day One free, or Notion AI) to document daily natural acceptances and violations. Discussion: Can AI ever have human values? Debate using structured arguments.

UNIT 2: Harmony in the Individual

Core Content: Harmony between Self and Body: Needs of the Self (happiness, trust, etc.) vs. Needs of the Body (food, shelter, clothing). Four Dimensions of Human Being: Icchha (desire), Vichar (thought), Anubhav (experience), Bodh (realisation). Feelings in Relationships: Nine feelings – Trust, Respect, Affection, Care, Guidance, Reverence, Glory, Gratitude, Love. Understanding Happiness and Prosperity: Holistic definition beyond materialistic view. Self-regulation vs. External Regulation: Inner motivation and its role in ethical conduct.

AI Integration: AI Tool: Replika (free AI companion) – explore conversations about emotional needs; critically reflect on AI limitations in understanding values. Activity: Create a 'Well-being Wheel' using Canva; use AI to identify gaps between material and value-based happiness. Journalling: Write a weekly reflection on one feeling (e.g., trust) experienced and observed.

UNIT 3: Harmony in the Family and Society

Core Content: Family as the Basic Unit of Living: Roles and responsibilities in family. Trust as the Foundational Value in Relationships. From Family to Society: Comprehensive Human Goal – Undivided Society. Universal Human Order: Five dimensions of human existence (Individual, Family, Society, Nature, Existence). Social Justice: Ensuring rights, duties, and dignified participation for all.

AI Integration: AI Tool: ChatGPT – generate scenarios of family conflicts and analyse through the lens of trust and respect. Case Study with AI: Use AI to map historical social movements (Gandhi, Ambedkar) onto the five-order framework. Project: Document 3 community service observations and analyse them using the harmony framework; use AI for structuring the report.

UNIT 4: Harmony in Nature and Existence

Core Content: Four Orders of Nature: Material, Plant/Bio, Animal, Human. Interconnectedness: Mutual fulfilment among the four orders. Ecological Balance: Current ecological crises and human responsibility. Sustainable Development Goals (SDGs) and Value-based Engineering. Right Understanding of Existence: Co-existence of units in space.

AI Integration: AI Tool: Global Forest Watch (open-source platform) – analyse deforestation data and correlate with lack of ecological values. AI Tool: ChatGPT – generate a sustainable engineering project idea and critique it using ecological harmony principles. Activity: Create an ecological audit of the campus using satellite images (Google Earth) and document AI-assisted analysis.

UNIT 5: Ethical Human Conduct and Professional Ethics

Core Content: Right Understanding, Right Feeling, Right Action: The value-ethics-skill triad. Professional Ethics: Codes of conduct for engineers (ASME, IEEE, IEI). Corruption, Dishonesty, and Short-term Thinking: Causes and value-based remedies. Vision for a Value-based Society and Universally Prosperous World. AI Ethics: Bias, fairness, transparency, and accountability in AI systems.

AI Integration: AI Tool: IBM AI Fairness 360 (open-source) – demonstrate AI bias using sample datasets. Case Study: Analyse Bhopal Gas Tragedy / Challenger Disaster through professional ethics lens. AI Ethics Audit: Use a given AI system (chatbot, recommendation engine) and evaluate it on fairness and transparency. Capstone: Write a 'Personal Value Charter' as a future engineer; use ChatGPT for drafting and self-editing for authenticity.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Recall and understand the concept of universal human values, natural acceptance, and the purpose of human life as articulated in Indian value philosophy.	L1/L2 (Remember/Understand)
CO2	Apply the framework of values-based living to personal relationships, family harmony, and societal well-being in daily life contexts.	L3 (Apply)
CO3	Analyse ethical dilemmas in engineering practice using the lens of universal human values and differentiate value-driven from interest-driven conduct.	L4 (Analyse)
CO4	Evaluate current engineering and social practices for their alignment with sustainability, ecological balance, and universal human order.	L5 (Evaluate)
CO5	Create a personal value charter and an action plan for ethical engineering practice using AI reflection tools and collaborative dialogue.	L6 (Create)

CO – PO Articulation Matrix

Scale: 3 = High, 2 = Medium, 1 = Low, – = No correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	-	-	-	-	-	2	2	3	1	2	-	2	1	1	2
CO2	-	-	-	-	-	3	2	3	2	2	1	2	1	1	2
CO3	1	2	-	1	-	3	2	3	2	2	1	3	1	2	2
CO4	1	2	2	2	2	3	3	3	2	2	2	3	2	2	2
CO5	1	2	2	1	2	3	2	3	3	3	2	3	2	2	2

Textbooks

1. Gaur, R.R., Sangal, R. & Bagaria, G.P. (2019). A Foundation Course in Human Values and Professional Ethics (3rd ed.). Excel Books, New Delhi.
2. Nagraj, A. (2015). Human Values and Professional Ethics: Universal Human Order. Jeevan Vidya Prakashan, Amarkantak.

Reference Books

1. Subramanian, C.V. (2020). Professional Ethics and Human Values. Oxford University Press.
2. Bajpai, Sarvesh Chandra. (2019). Engineering Ethics: Concepts and Cases. McGraw-Hill Education.
3. Flood, R.L. (2018). Rethinking the Fifth Discipline: Learning within the Unknowable. Routledge.
4. Covey, Stephen R. (2020). The 7 Habits of Highly Effective People (30th Anniversary ed.). Simon & Schuster.
5. Krishnamurti, J. (2018). Education and the Significance of Life. Harper Collins.
6. Singer, Peter. (2021). Practical Ethics (4th ed.). Cambridge University Press.

Web Resources & NPTEL Links

- AICTE – Value Education Resources: aicte-india.org
- UNESCO – Ethics of AI: www.unesco.org/en/artificial-intelligence/recommendation-ethics
- IEEE Ethics Resources: ethics.iit.edu
- NPTEL – Human Values & Professional Ethics: swayam.gov.in
- Jeevan Vidya Andolan: www.jeevanvidya.info
- YouTube: NPTEL – Human Values & Professional Ethics
- YouTube: TED-Ed – Ethics and Moral Philosophy
- YouTube: Michael Sandel – Justice: What's the Right Thing to Do?
- YouTube: AI Ethics Overview – IBM Technology
- YouTube: Crash Course – Philosophy

Recommended Free & Open-Source AI/Computational Tools

- ChatGPT / Claude (free tier) – for ethical-dilemma discussion, drafting and self-reflection exercises
- Notion AI / Day One (free tier) – AI journalling assistant for documenting natural acceptances
- Replika (free tier) – AI companion for exploring conversations about emotional needs
- Canva (free) – for creating well-being wheels and value-charter visuals
- Global Forest Watch (open-source) – for ecological/deforestation data analysis
- Google Earth (free) – for campus ecological audits using satellite imagery
- IBM AI Fairness 360 (open-source) – for demonstrating AI bias using sample datasets

II-Year I Sem

Course Code	MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS	L	T	Pr	P	Credits
26AMB02301T	Common to All Branches	2	0	0	0	2

Course Overview

Managerial Economics and Financial Analysis (MEFA) bridges economic theory with engineering practice, equipping students with decision-making tools for resource allocation, cost management, and financial planning. The course builds from foundational microeconomic concepts through to corporate financial analysis, emphasising practical application in engineering contexts. AI-powered tools—including data visualisation, forecasting models, and financial simulators—are embedded to enhance analytical capabilities and expose students to the growing intersection of economics, finance, and artificial intelligence.

Course Objectives

- Understand the fundamental concepts of managerial economics, demand analysis, and elasticity to make data-driven business decisions.
- Apply production and cost analysis techniques, including break-even analysis, to optimize organizational resource allocation.
- Analyze various forms of business organizations, market structures, and pricing strategies for competitive business environments.
- Evaluate capital budgeting proposals using NPV, IRR, ARR, and Payback methods for strategic investment decision-making.
- Prepare and interpret financial statements, compute key financial ratios, and employ AI/ML techniques for automated financial analysis.

Course Content

UNIT I: Introduction to Managerial Economics

Core Content: Managerial Economics – Meaning, Nature, Scope and significance for Engineers – Economic Analysis - Micro vs. Macroeconomics; positive vs. normative economics - Demand Analysis – Concept, Determinants, Demand Function, Law of Demand - Elasticity of Demand – Types, Measurement, Business Applications; Demand Forecasting – Meaning, Need, Methods.

Unit Outcomes: Explain the significance of managerial economics in digital and data-driven business environments. Identify the determinants of demand and construct a demand function for a given product. Compute and interpret price, income, and cross elasticities of demand with business applications. Apply demand forecasting methods and assess factors influencing forecast accuracy.

AI Integration: AI Tool: ChatGPT – generate demand scenarios for an engineering product (e.g., solar panels) and interpret price elasticity. AI Tool: Google Sheets + FORECAST function – implement Moving Average and Exponential Smoothing for a sample sales dataset. Activity: Use publicly available CMIE or RBI data; apply AI-assisted regression in Google Colab to forecast demand.

UNIT II: Production and Cost Analysis

Core Content: Production Concepts and Modern Manufacturing Systems - Production Function – Isoquants and Isocosts; Least-Cost Combination of Inputs; Cost Concepts – Fixed, Variable, Total, Average, Marginal Costs; Break-Even Analysis – Concept, Assumptions, Applications; Determination of Break-Even Point; Margin of Safety and Profit Planning (Simple problems).

Unit Outcomes: Illustrate production functions using isoquant and isocost diagrams. Determine the least-cost combination of inputs for optimal production decisions. Calculate the Break-Even Point and Margin of Safety for a given business scenario. Apply Break-Even Analysis for profit planning in manufacturing contexts.

AI Integration: AI Tool: Google Sheets / LibreOffice Calc (free) – build a BEA spreadsheet with dynamic charts. AI Tool: ChatGPT – generate a cost function for a manufacturing scenario and identify optimal production level. Activity: Collect cost data of a local small enterprise; perform AI-assisted CVP analysis and present findings. Data Tool: Plotly (open-source Python) or Google Data Studio – visualise cost curves.

UNIT III: Business Organisations and Market Structures

Core Content: Forms of Business Organisations – Sole Proprietary, Partnership, Joint Stock Companies, Startups and Digital Enterprises; Public vs Private Sector in Digital Economy; Market Structures – Classification of Markets with real-world examples - Pricing Methods and Strategies.

Unit Outcomes: Compare and contrast different forms of business organisations including startups and digital enterprises. Distinguish between public and private sectors in the context of the digital economy. Analyse characteristics of various market structures with real-world examples. Evaluate pricing strategies applicable to different market structures.

AI Integration: AI Tool: ChatGPT – simulate a duopoly pricing game and find Nash Equilibrium with AI assistance. Activity: Analyse pricing strategies of three competing tech products using online data and AI classification. Discussion: How do AI algorithms determine prices? Dynamic pricing in e-commerce (Amazon, Uber Surge Pricing).

UNIT IV: Capital Budgeting

Core Content: Capital – Introduction, Meaning, Nature and Significance - Capital Budgeting – Concept, Nature, Significance in Modern Enterprises, Role in Strategic Decision-Making; Methods – Payback Method, Accounting Rate of Return (ARR), Net Present Value (NPV), Internal Rate of Return (IRR) - Profitability Index (PI).

Unit Outcomes: Justify the importance of capital budgeting in strategic decision-making for engineering projects. Compute Payback Period and Accounting Rate of Return (ARR) for given investment proposals. Apply NPV and IRR techniques to evaluate and rank investment alternatives. Interpret capital budgeting results to support financial decisions in modern enterprises.

AI Integration: AI Tool: Google Sheets + NPV/IRR functions – evaluate 3 capital investment alternatives. Monte Carlo simulation for capital budgeting risk analysis. AI-based working capital optimization using predictive models. Decision tree analysis for investment appraisal using scikit-learn.

UNIT V: Financial Accounting and Analysis

Core Content: Accounting Principles and Digital Accounting Systems; Journal, Ledger, Trial Balance; Final Accounts – Trading Account, Profit and Loss Account, Balance Sheet with Simple Adjustments; Financial Analysis – Liquidity Ratios, Activity Ratios, Capital Structure Ratios, Profitability Ratios.

Unit Outcomes: Apply fundamental accounting principles in both traditional and digital accounting systems. Prepare journal entries, ledger accounts, and trial balance for simple business transactions. Construct Trading Account, Profit & Loss Account, and Balance Sheet with adjustments. Use financial ratio analysis to assess the overall financial health of an organisation.

AI Integration: Automated financial statement preparation using Python (pandas). AI Tool: ChatGPT – interpret a company's financial ratios from a pasted balance sheet and income statement. Machine learning for financial ratio prediction and company health assessment. Project: Download annual report of an Indian engineering company (BHEL, L&T, Infosys) and perform ratio analysis.

AI-Integrated Activities Summary

Unit	Activity	Description	AI Tool Used	Outcome Mapped
I	Demand Forecasting Simulation	Use AI tools to forecast demand for a consumer product using historical data	ChatGPT + Statista	CO1
I	Elasticity Chatbot Quiz	AI-generated adaptive quiz on types and measurement of elasticity	Google Gemini	CO1
II	Break-Even Calculator	Build a Break-Even model using Excel + Copilot for sensitivity analysis	Microsoft Copilot	CO2
II	Cost Curve Visualisation	Plot and interpret cost curves using AI-assisted data tools	Wolfram Alpha / Tableau	CO2
III	Market Structure Case Study	Generate real-world case studies of oligopoly (e.g., Telecom sector) using AI	Claude AI / ChatGPT	CO3

III	Startup Evaluation Report	Evaluate a digital startup's business model using AI-generated insights	Google Gemini	CO3
IV	Capital Budgeting Decision Tool	Use AI to compare NPV, IRR for two competing engineering projects	Copilot + Wolfram Alpha	CO4
IV	IRR Sensitivity Analysis	Explore how changes in cash flows affect IRR using AI-powered spreadsheets	Microsoft Copilot	CO4
V	Digital Accounting Exercise	Prepare journal, ledger, and trial balance using Tally AI / Zoho Books	Tally AI / Zoho AI	CO5
V	Financial Ratio Dashboard	Construct a financial ratio dashboard from a company's annual report using Tableau AI	Tableau AI	CO5

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Recall and understand fundamental concepts of economics including demand, supply, market structures, and the time value of money.	L1/L2 (Remember/Understand)
CO2	Apply economic theories of demand forecasting, cost analysis, and production functions to engineering project decision-making scenarios.	L3 (Apply)
CO3	Analyse financial statements, cost-volume-profit relationships, and capital budgeting decisions using traditional and AI-powered financial tools.	L4 (Analyse)
CO4	Evaluate investment alternatives using NPV, IRR, and Payback Period methods and assess business viability under uncertainty using scenario analysis.	L5 (Evaluate)
CO5	Create a comprehensive mini-project business plan incorporating market analysis, cost estimation, financial projections, and AI-assisted forecasting.	L6 (Create)

CO – PO Articulation Matrix

Scale: 3 = High, 2 = Medium, 1 = Low, – = No correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	2	1	1	1	2	1	1	1	2	2	2	1	1	2
CO2	2	3	2	2	2	2	1	1	1	2	3	2	2	2	2
CO3	2	3	2	3	3	2	1	1	2	2	3	3	2	2	1
CO4	2	3	2	3	3	2	1	1	2	2	3	3	2	2	1
CO5	2	3	3	3	3	2	2	1	3	3	3	3	3	3	1

PO Descriptions for Reference: PO1 Engineering Knowledge | PO2 Problem Analysis | PO3 Design/Development | PO4 Investigation | PO5 Modern Tool Usage | PO6 Engineer & Society | PO7 Environment & Sustainability | PO8 Ethics | PO9 Individual & Teamwork | PO10 Communication | PO11 Project Management | PO12 Life-long Learning. PSO1: Apply economics and financial analysis principles to engineering and management contexts.

Textbooks

1. Mithani, D.M. Managerial Economics: Theory and Applications. Himalaya Publishing House, 10th Edition, 2025.
2. Pandey, I.M. Financial Management. Vikas Publishing House, New Delhi, 12th Edition, 2022.
3. Chandramouli Subramanian & Asha A George. Artificial Intelligence: Principles and Applications. Universities Press Private Limited, Hyderabad, First Edition, 2026.

Reference Books

1. Aryasri, A.R. Managerial Economics and Financial Analysis. McGraw-Hill Education, Revised Edition, 2020.
2. Prasanna Chandra. Financial Management: Theory and Practice. Tata McGraw-Hill, New Delhi, 10th Edition, 2019.
3. Kumar, U.D. Business Analytics: The Science of Data-Driven Decision Making. Wiley India, 2nd Edition, 2021.
4. Siddiqui, S. A. Engineering Economics and Financial Accounting. S. Chand Publishing, New Delhi, 3rd Edition, 2019.
5. Gupta, G. S. Economics for Engineers. Tata McGraw-Hill Education, 2nd Edition, 2017.

Web Resources & NPTEL Links

- NPTEL (IIT) – Managerial Economics: https://nptel.ac.in/courses/110105063

- NPTEL (IIT) – Financial Accounting for Engineers: https://nptel.ac.in/courses/110107114
- Coursera – Economics for Non-Economists (Univ. of California): https://www.coursera.org/learn/economics
- edX – Financial Accounting (MIT OpenCourseWare): https://www.edx.org/course/financial-accounting
- Khan Academy – Microeconomics & Finance Basics: https://www.khanacademy.org/economics-finance-domain
- SWAYAM Portal – Business Economics (UGC-approved): https://swayam.gov.in

Recommended Free & Open-Source AI/Computational Tools

- ChatGPT / Claude AI – Concept clarification, case study generation, problem solving assistance
- Microsoft Copilot – Financial analysis, spreadsheet automation, data interpretation
- Google Gemini – Research support, market structure comparisons, summarisation
- Tally AI / Zoho Books AI – Digital accounting simulations, ledger preparation, journal entries
- Statista / Tableau AI – Data visualisation for demand trends and financial ratios
- Wolfram Alpha – Quantitative problems – NPV, IRR, Break-Even, elasticity calculations Grammarly AI – Report writing, case study docum

II Year I Sem

ANALOG CIRCUITS

Course Code	Year/Sem	L	T	Pr	P	Credits
26AEC04304T		3	0	0	0	3

Prerequisites: Engineering Physics

Course Objectives

- Understand PN junction diode, rectifier circuits, Zener regulation, and BJT biasing methods with Q-point analysis.
- Understand negative feedback amplifier topologies and study RC phase shift, Wien bridge, and crystal oscillators using the Barkhausen criterion.
- Understand ideal and practical Op-Amp characteristics and analyse inverting, non-inverting, integrator, differentiator, and log/antilog amplifier circuits.
- Study comparators, waveform generators, 555 timer monostable/astable circuits, Schmitt trigger, and PLL block operation and applications.
- Understand weighted resistor and R-2R ladder DAC techniques and evaluate flash, counter-type, SAR, and dual slope ADC types for speed, resolution, and accuracy.

Course Content

UNIT I: PN Junction Diode, Applications & BJT

Core Content: PN junction diode, VI Characteristics, Diode resistance, Transition and Diffusion Capacitance, Quantitative analysis of Half-wave, Full-wave and Bridge Rectifiers and Capacitor filter, Zener diode as Voltage Regulator. Bipolar Junction Transistors: Characteristics, Transistor as an Amplifier and as a Switch, BJT Configurations, Biasing and Stabilization: Operating Point, DC and AC Load Lines, Importance of Biasing, Self-Bias, Bias Stability, Thermal Runaway, Illustrative problems.

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain how a PN junction diode works in forward and reverse bias; ask it to solve one numerical on the diode current equation and verify manually. Also ask it to compare BJT Configurations.
2. Use Python (Matplotlib) or an online simulator (Falstad / EveryCircuit) to plot the diode V-I characteristic and observe how it changes with temperature; then simulate a full-wave bridge rectifier with and without a filter capacitor and note how capacitor value affects ripple voltage.
3. Use an LLM to walk through the DC load line construction and Q-point determination for a given self-bias circuit; verify the operating point (VCE, IC) by solving the bias equations manually for one set of component values.

Free/Open-Source AI Tools: AI/LLM tool; Python (Matplotlib); Falstad; EveryCircuit.

UNIT II: Feedback Amplifiers & Oscillator Circuits

Core Content: Feedback Amplifiers: Classification of Amplifiers, the Feedback Concept, General Characteristics of Negative-Feedback Amplifiers, Effect of Negative Feedback upon

Output and Input Resistances, Voltage-Series Feedback, Current-Series Feedback, Current-Shunt Feedback, Voltage-Shunt Feedback. Oscillator Circuits: Barkhausen Criterion of oscillation, Oscillator operation, Hartley oscillator, Colpitts oscillator, RC phase shift oscillator, Wien bridge Oscillator, Crystal Oscillator. (Quantitative analysis only for all the feedback amplifiers and oscillators)

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain the four negative feedback topologies (voltage-series, current-series, voltage-shunt, current-shunt); ask it to give one circuit example for each and describe how each changes the input and output resistance — prepare a brief comparison table.
2. Use an AI/LLM tool to explain the Barkhausen criterion; ask it to show why both conditions (loop gain = 1 and phase shift = 0° or 360°) must hold simultaneously, using the RC phase shift oscillator as an example — verify the frequency formula $f = 1/(2\pi\sqrt{6}\ RC)$ for one set of R and C values.
3. Use Python or Falstad (online simulator) to simulate an RC phase shift oscillator for a target frequency of 1 kHz; vary R and C values, observe the output frequency change, and record at least three combinations in a table to verify the formula.

Free/Open-Source AI Tools: AI/LLM tool; Python; Falstad.

UNIT III: Operational Amplifiers & Applications

Core Content: Operational Amplifiers: Introduction, Basic information of Op-Amp, Ideal Operational Amplifier, Block Diagram Representation of Typical Op-Amp, DC and AC characteristics, 741 Op-Amp and its features. Op-Amp Applications: Basic Op-Amp Applications, Instrumentation Amplifier, AC Amplifier, V-to-I and I-to-V Converter, Sample and Hold Circuit, Log and Antilog Amplifier, Multiplier and Divider, Differentiator, Integrator.

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain ideal Op-Amp assumptions and how they break down in the real 741 IC (slew rate, CMRR, finite gain); then ask it to derive the output expression for an inverting amplifier and a summing amplifier using the virtual ground concept — verify by substituting values.
2. Use LTspice or Falstad to simulate an Op-Amp integrator and differentiator with a square wave input; observe the output waveforms and write one paragraph explaining why the output looks the way it does, relating it to the RC time constant.
3. Use Python (NumPy / Matplotlib) to plot the output of a log amplifier (Vout vs. Vin) for inputs from 0.1 V to 2 V; compare the plot with the theoretical expression and identify the voltage range where it holds well — note any deviation at high input voltages.

Free/Open-Source AI Tools: AI/LLM tool; LTspice; Falstad; Python (NumPy, Matplotlib).

UNIT IV: Comparators, Waveform Generators, Timers & PLL

Core Content: Comparators and Waveform Generators: Introduction, Comparator, Square Wave Generator, Monostable Multivibrator, Triangular Wave Generator, Sine Wave Generators. Timers and Phase Locked Loop: Introduction to 555 timer, functional diagram, Monostable and Astable operations and applications, Schmitt Trigger, PLL block schematic, principles and description of individual blocks, 565 PLL, Applications of VCO (566).

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain the 555 timer in monostable and astable modes — ask it to describe how the internal comparators and flip-flop produce the timing waveform and what changes between the two modes; verify the formulas T = 1.1RC (monostable) and f = 1.44/((R1+2R2)C) (astable) for one set of values.
2. Use an online simulator (Falstad / EveryCircuit) or LTspice to build a 555 astable circuit for 1 kHz output; vary R1, R2, and C to change the duty cycle and record at least three combinations in a table showing frequency and duty cycle.
3. Use an AI/LLM tool to explain the Phase Locked Loop (PLL) — describe the role of the phase detector, low-pass filter, and VCO in simple terms; ask it to give one practical application (e.g., FM demodulation or frequency synthesis) with a brief block-level description.

Free/Open-Source AI Tools: AI/LLM tool; Falstad; EveryCircuit; LTspice.

UNIT V: DAC and ADC Converters

Core Content: Digital to Analog and Analog to Digital Converters: Introduction, basic DAC techniques, weighted resistor DAC, R-2R ladder DAC, inverted R-2R DAC. A-D Converters — parallel comparator type ADC, counter type ADC, successive approximation ADC and dual slope ADC. DAC and ADC Specifications.

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain the difference between a weighted resistor DAC and an R-2R ladder DAC; ask it to explain why the R-2R ladder is preferred in practice — verify the output voltage formula for a 4-bit R-2R DAC with Vref = 5 V manually for two input codes.
2. Use an AI/LLM tool to compare the four ADC types (flash, counter-type, successive approximation, dual slope); ask it to fill a comparison table covering conversion speed, hardware complexity, accuracy, and one typical application for each — review and correct the table yourself.
3. Use Python to simulate a 4-bit R-2R ladder DAC: calculate the analog output for all 16 input codes (0000–1111) with Vref = 5 V and plot the staircase output using Matplotlib; then manually trace the SAR algorithm for one input (e.g., 3.2 V) and verify the digital output.

Free/Open-Source AI Tools: AI/LLM tool; Python; Matplotlib.

Instructions:

- The suggested tasks can be modified based on the decision of the subject expert.
- The suggested tasks can be part of assignments using appropriate AI/LLM tools and Python/MATLAB-based simulations, with the choice of platforms and machine learning models decided by the subject expert to enhance conceptual understanding and practical AI-enabled engineering skills.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Analyse diode and Zener characteristics, evaluate rectifier performance, and design BJT bias circuits for Q-point stability.	Analyse

CO	Course Outcome Statement	Bloom's Level
CO2	Evaluate negative feedback topologies for gain and impedance, and verify oscillation frequency using Barkhausen criterion for RC phase shift and Wien bridge oscillators.	Evaluate
CO3	Evaluate 741 Op-Amp characteristics and design inverting, non-inverting, integrator, differentiator, log/antilog, and instrumentation amplifier circuits.	Evaluate
CO4	Analyse comparator, 555 timer, and Schmitt trigger circuits — computing pulse width, frequency, and duty cycle — and explain PLL operation and applications.	Analyse
CO5	Compare weighted resistor and R-2R ladder DAC techniques and evaluate flash, counter-type, SAR, and dual slope ADC types for speed, resolution, and accuracy.	Compare

CO – PO Articulation Matrix

Note: 1 – Low, 2 – Medium, 3 – High, '-' – No Correlation

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	-	-	2	-	-	-	-	-	1	1	3	2
CO2	3	3	2	-	2	-	-	-	-	-	1	2	3	2
CO3	2	3	3	2	3	-	-	1	-	-	2	3	3	2
CO4	2	2	3	3	3	1	1	1	1	-	2	3	3	2
CO5	2	2	3	3	3	1	1	2	2	1	3	3	3	2

Textbooks

1. J. Millman, C. Halkias, "Electronic Devices and Circuits", Tata McGraw-Hill, 2nd Edition, 2010.
2. D. Roy Choudhury, "Linear Integrated Circuits", New Age International (P) Ltd, 2nd Edition, 2003.

Reference Books

1. Robert L. Boylestad and Louis Nashelsky, "Electronic Devices and Circuit Theory", Pearson Education, 2021.
2. G. K. Mithal, "Electronic Devices and Circuits", Khanna Publisher, 23rd Edition, 2017.

3. David Bell, "Electronic Devices and Circuits", Oxford University Press, 5th Edition, 2008.
4. Malvino, Albert Paul, and David J. Bates, "Electronic Principles", McGraw-Hill Higher Education, 2007.
5. R. A. Gayakwad, "Operational Amplifiers and Linear Integrated Circuits", Prentice Hall India, 2002.

Web Resources & NPTEL Links

- NPTEL Course: https://nptel.ac.in/courses/122106025
- NPTEL Course: https://nptel.ac.in/courses/108102112

II Year I Sem

Course Code	AI ASSISTED ELECTRICAL CIRCUIT ANALYSIS	L	T	Pr	P	Credits
26AEE02301T	II Year I Sem	2	0	2	0	3
Pre-requisite	Differential Equations & Vector Calculus, AI Tools & Applications					

Course Objectives

- To introduce the fundamental concepts of electrical circuits and analyze DC and AC circuits using basic network laws and analysis techniques.
- To apply network theorems and resonance concepts for solving complex electrical circuit problems.
- To analyze balanced and unbalanced three-phase circuits and power measurement techniques.
- To study transient behavior of electrical circuits using differential equation and Laplace transform methods.
- To understand two-port network parameters and apply AI-based computational tools for circuit analysis and optimization.

Course Content

UNIT I: Introduction to Electrical Circuits

Core Content: Basic Concepts of passive elements of R, L, C and their V-I relations; Sources (dependent and independent); Ohm's Law and its limitations; Kirchhoff's laws; Network reduction techniques (series, parallel, series-parallel, star-to-delta and delta-to-star transformation); source transformation technique; node and mesh analysis. Single Phase Circuits: Characteristics of periodic functions; Average value, R.M.S. value, form factor; representation of a sine function; concept of phasor, phasor diagrams. Steady state analysis of R, L and C circuits to sinusoidal excitations; response of pure resistance, inductance, capacitance, series RL circuit, series RC circuit, series RLC circuit, parallel RL circuit, parallel RC circuit.

AI Integration: AI-Assisted Circuit Analysis and Error Detection: Use AI tools (Python + machine learning) to automatically verify KCL/KVL equations, detect incorrect circuit connections, and assist students in solving node and mesh analysis problems.

Free/Open-Source AI Tools: *Python (NumPy, SciPy, Matplotlib) on Google Colab; Ngspice (free); Scikit-learn; Jupyter Notebook*

UNIT II: Network Theorems (DC & AC Excitations)

Core Content: Superposition theorem; Thevenin's theorem; Norton's theorem; Maximum Power Transfer theorem; Reciprocity theorem; Millman's theorem; Compensation theorem. Resonance and Locus Diagrams: Series Resonance: Characteristics of a series resonant circuit; Q-factor, selectivity and bandwidth; expression for half power frequencies; Parallel resonance: Q-factor, selectivity and bandwidth; Locus diagram: Series RL, RC, RLC with R, L and C variables.

AI Integration: Machine Learning for Equivalent Circuit Prediction: Use regression or neural network models to estimate Thevenin/Norton equivalents and predict resonance characteristics (Q-factor and bandwidth) from circuit parameters.

Free/Open-Source AI Tools: *Python (NumPy, SciPy, Scikit-learn) on Google Colab; Ngspice; Jupyter Notebook*

UNIT III: Analysis of Three Phase Balanced Circuits

Core Content: Phase sequence; star and delta connection of sources and loads; relation between line and phase quantities; analysis of balanced three phase circuits; measurement of active and reactive power. Analysis of three phase unbalanced circuits: Loop method; Star-Delta transformation technique; two wattmeter method for measurement of three phase power.

AI Integration: AI-Based Fault Detection in Three-Phase Systems: Application of machine learning algorithms to detect phase imbalance, faults, and abnormal power consumption patterns in three-phase electrical systems using voltage and current data.

Free/Open-Source AI Tools: *Python (Scikit-learn, Pandas) on Google Colab; Ngspice; Jupyter Notebook*

UNIT IV: Transient Analysis

Core Content: Transient response of R-L, R-C and R-L-C circuits (Series and parallel combinations) for D.C. and sinusoidal excitations; Initial conditions; Solution using differential equation approach and Laplace transform approach.

AI Integration: AI Integration to Transient Analysis: Use regression models to predict voltage and current transients in RL, RC, and RLC circuits after switching.

Free/Open-Source AI Tools: *Python (NumPy, SciPy, Matplotlib) on Google Colab; Ngspice; Jupyter Notebook*

UNIT V: Network Parameters

Core Content: Impedance parameters; Admittance parameters; Hybrid parameters; Transmission (ABCD) parameters; conversion of Parameters from one form to other; Conditions for Reciprocity and Symmetry; Interconnection of Two Port networks in Series, Parallel and Cascaded configurations; problems.

AI Integration: AI Optimization of Two-Port Network Parameters: Use AI optimization techniques (genetic algorithms or neural networks) to determine optimal two-port network parameters.

Free/Open-Source AI Tools: *Python (NumPy, SciPy, Scikit-learn) on Google Colab; Ngspice; Scikit-learn; Jupyter Notebook*

Course Outcomes (COs)		
CO	**Course Outcome Statement**	**Bloom's Level**
CO1	Remember and explain the fundamental concepts of electrical circuit elements, network laws, resonance, transient response, three-phase circuits, and two-port network parameters used in circuit analysis.	L1 (Remember) L2 (Understand)
CO2	Apply Ohm's law, Kirchhoff's laws, network reduction techniques, node and mesh analysis, network theorems, resonance conditions, and transient analysis methods to solve DC and AC electrical circuit problems.	L3 (Apply)
CO3	Analyze the performance of balanced and unbalanced three-phase circuits, resonant circuits, transient responses, and interconnected two-port networks under different operating conditions.	L4 (Analyze)
CO4	Apply and analyze AI and machine learning techniques for automated circuit verification, fault detection, resonance	L3 (Apply) L4 (Analyze)

	prediction, transient estimation, and optimization of electrical network parameters using open-source simulation tools.	
CO5	Design electrical circuit models and AI-assisted solutions for equivalent circuit estimation, three-phase system monitoring, transient response prediction, and optimization of two-port network configurations using Python, Ngspice, and Scikit-learn.	L4 (Analyze) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	L1, L2	3	2	1	1	2	1	–	–	1	–	2	3	1	1
CO2	L3	3	3	2	2	2	1	–	–	1	–	2	3	2	1
CO3	L4	3	3	2	2	2	1	–	–	1	–	2	3	2	1
CO4	L3, L4	3	3	2	3	3	1	–	–	1	–	2	3	2	2
CO5	L4, L6	3	2	3	2	3	1	–	–	2	1	3	2	3	3

Textbooks

1. Engineering Circuits Analysis, Jack Kemmerly, William Hayt and Steven Durbin, Tata McGraw Hill Education, 2005, Sixth Edition.

2. Fundamentals of Electric Circuits, Charles K. Alexander, Mathew N. O. Sadiku, 3rd Edition, Tata McGraw-Hill, 2019.

Reference Books

1. Network Analysis, M. E. Van Valkenburg, 3rd Edition, PHI, 2019.

2. Network Theory, N. C. Jagan and C. Lakshminarayana, 1st Edition, B. S. Publications, 2012.

3. Circuits and Networks Analysis and Synthesis, A. Sudhakar, Shyam Mohan S. Palli, 5th Edition, Tata McGraw-Hill, 2017.

4. Engineering Network Analysis and Filter Design, Durgesh C. Kulshreshtha, Gopal G. Bhise, Prem R. Chadha, Umesh Publications, 2012.

5. Circuit Theory: Analysis and Synthesis, A. Chakrabarti, Dhanpat Rai & Co., 2018, 7th Revised Edition.

Web Resources & NPTEL Links

- NPTEL – Electrical Circuits: https://archive.nptel.ac.in/courses/117/106/117106108/
- NPTEL – Circuit Theory: https://archive.nptel.ac.in/courses/108/105/108105159/
- MIT OpenCourseWare – Circuits and Electronics: https://ocw.mit.edu

Recommended Free & Open-Source AI/Computational Tools

Python (NumPy, SciPy, Scikit-learn) – FREE on Google Colab for circuit analysis automation
Ngspice (free, open-source) – Circuit simulation
Scikit-learn (free, open-source) – Machine learning models for circuit parameter estimation
TensorFlow / Keras (free, open-source) – Neural network models
Jupyter Notebook (free, open-source) – Interactive computational environment

II Year I Sem

Course Code	DC MACHINES AND TRANSFORMERS WITH INTELLIGENT CONTROL	L	T	Pr	P	Credits
26AEE02302T	II Year I Sem	2	0	2	0	3
Pre-requisite	Electromagnetic Field Theory, AI Tools & Applications, Electrical Circuits					

Course Objectives

- To understand the construction, working principle and performance characteristics of DC generators and their practical applications.
- To analyze the operating characteristics and performance of different types of DC motors and their starting mechanisms.
- To study speed control methods and testing procedures of DC machines using analytical and computational approaches.
- To understand the construction, operation and performance characteristics of single-phase transformers.
- To analyze testing methods, efficiency and parallel operation of transformers using modern computational and AI-based techniques.

Course Content

UNIT I: DC Generators

Core Content: Construction and principle of operation of DC machines; EMF equation of DC generators; Excitation techniques; Armature reaction and commutation; Methods of improving commutation; Open circuit characteristics and load characteristics of DC generators; Parallel operation of DC generators; Applications of DC generators.

AI Integration: AI-based prediction of generated EMF and load characteristics; AI-assisted analysis of parallel operation performance; Data visualization of generator characteristics; Predictive maintenance of DC generators using sensor data.

Free/Open-Source AI Tools: *Python (NumPy, Pandas, SciPy); Scikit-learn; TensorFlow/PyTorch; GNU Octave/Scilab*

UNIT II: DC Motors

Core Content: DC motor – Back-emf; Generation of torque; Torque equation; Performance characteristics of different types of motors; Necessity of starters; 3-point and 4-point starters; Losses in DC machines; Applications of DC Motors; numerical problems.

AI Integration: AI models for predicting torque and speed characteristics; Intelligent starter protection and monitoring; Machine learning models for efficiency estimation; Intelligent control systems for motor performance optimization.

Free/Open-Source AI Tools: *Python (SciPy, NumPy); Scikit-learn; OpenModelica; TensorFlow*

UNIT III: Speed Control and Testing of DC Machines

Core Content: Speed control of DC Machines: Armature control and Flux control methods; Testing of DC machines: Brake test; Swinburne's test; Hopkinson's test; Fields test; Retardation test.

AI Integration: AI-assisted speed control optimization of DC motors; AI-assisted virtual experimentation and result analysis; AI-based performance testing and fault prediction; Simulation of DC motor speed control methods.

Free/Open-Source AI Tools: *Python (Pandas, Matplotlib); Scikit-learn; OpenModelica; Jupyter Notebook*

UNIT IV: Single-Phase Transformers

Core Content: Introduction to single-phase Transformers (Construction and principle of operation); EMF equation; operation on no-load and on load; lagging, leading and unity power factors loads; phasor diagrams; equivalent circuit; regulation; losses and efficiency; effect of variation of frequency and supply voltage on losses; all day efficiency; Applications.

AI Integration: Machine learning prediction of transformer efficiency and losses; AI-based load forecasting for transformer operation; Neural networks for transformer voltage regulation prediction; Visualization of transformer operating characteristics.

Free/Open-Source AI Tools: *Python (NumPy, SciPy); TensorFlow/PyTorch; OpenModelica; ParaView*

UNIT V: Testing of Transformers

Core Content: Open Circuit and Short Circuit tests; Sumpner's test; separation of losses; Parallel operation with equal and unequal voltage ratios; auto transformer; equivalent circuit; comparison with two winding transformers; Scott connection; off load and on load tap changers.

AI Integration: Data analytics for separation of hysteresis and eddy current losses; AI-assisted modeling of load variation and all-day efficiency; Data-driven models for loss separation and efficiency prediction; AI optimization of transformer load sharing.

Free/Open-Source AI Tools: *Python (Scikit-learn, Pandas); TensorFlow; OpenModelica; Elmer FEM*

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the construction, principle of operation, characteristics, and applications of DC generators, DC motors, and single-phase transformers.	L1 (Remember) L2 (Understand)
CO2	Understand the concepts of armature reaction, commutation, speed control methods, transformer equivalent circuits, regulation, and testing procedures.	L2 (Understand)
CO3	Apply testing methods, performance analysis techniques, and simulation tools for evaluating DC machines and transformers.	L3 (Apply)
CO4	Analyze the efficiency, losses, regulation, speed characteristics, and operational performance of electrical machines using conventional and AI-based techniques.	L4 (Analyze)
CO5	Develop AI-assisted models for prediction, visualization, and performance evaluation of electrical machines and transformers using modern open-source tools.	L4 (Analyze) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3

CO1	L1, L2	3	2	1	1	2	1	–	–	1	–	2	3	1	1
CO2	L2	3	3	2	2	2	1	–	–	1	–	2	3	2	1
CO3	L3	3	3	2	3	3	1	–	–	1	1	2	3	2	2
CO4	L4	3	2	2	2	3	1	–	–	1	–	2	3	2	2
CO5	L4, L6	3	3	2	3	3	1	–	–	2	1	3	2	3	3

Textbooks

1. Electrical Machinery by Dr. P S Bimbhra, 7th Edition, Khanna Publishers, New Delhi, 1995.
2. Performance and Analysis of AC Machines by M.G. Say, CBS, 2002.

Reference Books

1. Electrical Machines by D. P. Kothari, I. J. Nagarth, McGraw Hill Publications, 5th Edition.
2. Electrical Machinery Fundamentals by Stephen J Chapman, McGraw Hill Education, 2011.
3. Generalized Theory of Electrical Machines by Dr. P S Bimbhra, 7th Edition, Khanna Publishers, 2021.
4. Theory & Performance of Electrical Machines by J. B. Gupta, S. K. Kataria & Sons, 2007.
5. Electric Machinery by Fitzgerald, A. E. Kingsley Jr. C., & Umans S. D., 7th Edition, McGraw-Hill Education, 2014.

Web Resources & NPTEL Links

• NPTEL – Electrical Machines I: https://nptel.ac.in/courses/108105017
• NPTEL – Transformer Course: https://nptel.ac.in/courses/108105155
• MIT OpenCourseWare – Electric Machines: https://ocw.mit.edu/courses/electric-machines/
• Python for Electrical Engineering: https://scipy.org
• OpenModelica Simulation: https://openmodelica.org

Recommended Free & Open-Source AI/Computational Tools

Python (NumPy, Pandas, SciPy) – FREE on Google Colab for machine analysis
Scikit-learn (free, open-source) – Machine learning models
TensorFlow / PyTorch (free, open-source) – Predictive modelling
GNU Octave / Scilab (free) – Electrical machine simulation
OpenModelica (free, open-source) – Electrical machine simulation
Elmer FEM (free, open-source) – Finite element modelling

II Year - I Sem.

ANALOG CIRCUITS LAB

Course Code	Year/Sem	L	T	Pr	P	Credits

Course Code	Year/Sem	L	T	Pr	P	Credits
26AEC04304P		0	0	0	2	1

Prerequisites: Analog Circuits

Course Objectives

- To understand the characteristics and operation of semiconductor devices, rectifiers, transistor configurations, oscillators, timers, Op-Amp circuits, and DACs using laboratory experiments.
- To explain the working principles of electronic circuits and interpret experimental waveforms, biasing conditions, oscillations, amplification, and digital-to-analog conversion processes.
- To apply electronic circuit theory for designing, constructing, and testing analog electronic circuits using BJTs, Op-Amps, 555 timers, oscillators, and DACs.
- To analyse the performance of electronic circuits through measurements, graphical interpretation, comparison of theoretical and practical results, and simulation using AI/LLM, Python, or MATLAB tools.
- To design and evaluate AI-assisted and simulation-based electronic system models for prediction, optimisation, regression analysis, and validation of laboratory experiments.

List of Practices

Implement / execute any 12 of the following experiments:

1. Plot V–I characteristics of a PN Junction diode and determine static and dynamic resistance.
2. Plot V–I characteristics of a Zener diode and verify its functionality as a voltage regulator.
3. Implement half-wave and full-wave bridge rectifiers with and without filter; measure ripple factor and rectification efficiency.
4. Verify the input and output characteristics of BJT in CE configuration; determine β, find Q-point on DC load line, and identify active, saturation, and cut-off regions.
5. Verify the input and output characteristics of BJT in CB configuration; determine α and compare with CE results.
6. Design and analyse a voltage-divider (self-bias) circuit using BJT; verify the Q-point (VCE, IC) experimentally and compare with the designed values.
7. Construct and test Hartley oscillator; measure the frequency of oscillation, verify against the theoretical formula, and observe sustained sinusoidal output on CRO.
8. Construct and test Colpitts oscillators; measure the frequency of oscillation, verify against the theoretical formula, and observe sustained sinusoidal output on CRO.
9. Design and observe the output of an RC Phase Shift Oscillator using transistor or Op-Amp; verify $f = 1/(2\pi\sqrt{6}\,RC)$ and measure the phase shift across each RC stage.
10. Construct an Astable Multivibrator using 555 timer; design for a specified frequency and duty cycle using $f = 1.44/((R1+2R2)C)$; verify on CRO and vary R2 to observe duty cycle change.
11. Design and verify a Monostable Multivibrator using 555 timer; design for a specified pulse width $T = 1.1RC$; trigger with external pulse and verify output pulse width on CRO.

12. Design and implement Op-Amp applications — inverting summing amplifier (adder), difference amplifier (subtractor), and open-loop comparator using IC 741; verify output for given inputs.
13. Design and verify Op-Amp Integrator and Differentiator circuits using IC 741; apply square wave to integrator (observe triangular output) and triangular wave to differentiator (observe square output).
14. Construct and verify an R-2R Ladder DAC; apply all 16 binary input combinations (0000–1111) and measure analog output; verify step size = $Vref/2^n$ and plot staircase output.

AI Integration – AI/LLM and/or Python/MATLAB Tasks

Any six experiments need to be implemented using appropriate AI/LLM tools and/or Python/MATLAB based simulations:

1. (a) Use an AI/LLM tool to interpret the forward bias, reverse bias, and breakdown regions of the V-I curve and explain the significance of static resistance, dynamic resistance, and reverse saturation current. (b) Train a regression model to fit the Shockley diode equation $I = I_0(e^{(V/\eta VT)} – 1)$ to the measured V-I data and estimate the saturation current and ideality factor.
2. (a) Use an AI/LLM tool to explain filter action and ripple reduction in rectifiers, describe the effect of capacitor value on ripple, and derive expressions for ripple factor and rectification efficiency. (b) Use Python/MATLAB simulation to predict ripple voltage for different capacitor values; apply regression to model ripple factor as a function of load resistance and capacitance.
3. (a) Use an AI/LLM tool to explain the physical significance of each region (active, saturation, cut-off) of the CE characteristics, the concept of current gain β, and the Early effect. (b) Train a linear regression model in Python to estimate β from the measured IC vs. IB data and evaluate model accuracy against graphically extracted values.
4. (a) Use an AI/LLM tool to reason about the relationship between α and β, explain the low input impedance and near-unity current gain of CB configuration, and compare it with CE. (b) Use Python regression to predict IC from IB using the measured CB characteristic data and evaluate model accuracy.
5. (a) Use an AI/LLM tool to explain the voltage divider bias technique, the DC load line concept, the role of each component (R1, R2, RC, RE) in bias stabilisation, and how the Q-point ensures linear amplifier operation. (b) Build a Python regression-based Q-point predictor that takes R1, R2, RC, RE, and VCC as inputs and predicts VCE and IC.
6. (a) Use an AI/LLM tool to explain why the CC configuration has near-unity voltage gain and high input impedance, describe the buffer action, and compare its characteristics with the CE amplifier. (b) Use Python to simulate the emitter follower frequency response and compare gain, input impedance, and output impedance with the CE amplifier using measured data.
7. (a) Use an AI/LLM tool to explain the Barkhausen criterion, the role of the LC tank circuit in sustaining oscillations, and derive the theoretical oscillation frequency formula for both Hartley and Colpitts configurations. (b) Use Python to compute and compare theoretical vs. measured oscillation frequencies for both oscillators and build a regression model to predict frequency from L and C values.
8. (a) Use an AI/LLM tool to explain the 180° phase shift condition across the three RC stages, the loop gain requirement for sustained oscillations, and derive the frequency formula $f = 1/(2\pi\sqrt{6}\, RC)$. (b) Use Python to plot the expected oscillation frequency vs. RC values and verify the measured result against the theoretical curve; train a regression model to predict frequency from R and C values.

9. (a) Use an AI/LLM tool to explain the internal working of the 555 timer in astable mode — including the comparators, flip-flop, threshold/trigger levels — and derive the frequency and duty cycle formulas. (b) Use Python to plot frequency and duty cycle as a function of R2 for a fixed C and compare with measured CRO values; apply regression to predict frequency from component values.
10. (a) Use an AI/LLM tool to explain the difference between monostable and astable modes in the 555 timer and derive the pulse width formula T = 1.1RC by analysing the capacitor charging curve. (b) Use Python to simulate the capacitor charging curve and verify the derived pulse width formula T = 1.1RC; train a regression model to predict pulse width from R and C values.
11. (a) Use an AI/LLM tool to derive and explain the output expressions for the inverting summing amplifier and difference amplifier using the virtual ground concept, and describe open-loop comparator operation. (b) Use Python to simulate output voltage vs. input combinations for all three circuits and verify all cases of the comparator truth table; apply regression to validate amplifier gain accuracy.
12. (a) Use an AI/LLM tool to explain the mathematical basis of integration and differentiation performed by Op-Amp RC circuits and relate the square/triangular waveform transformation to the RC time constant. (b) Use Python (NumPy) to numerically integrate and differentiate the input waveforms and compare the results with CRO observations; apply regression to estimate the RC time constant from the output slope.
13. (a) Use an AI/LLM tool to explain the principle of binary-weighted voltage addition in the R-2R network, derive the output voltage expression Vout = (Vref/2^n) × digital code, and discuss quantisation error. (b) Use Python to simulate DAC output for all 16 input codes, plot the staircase waveform, and calculate percentage error between ideal and measured outputs; apply regression to model linearity error.

Instructions:

- The suggested AI/LLM and/or Python/MATLAB tasks can be modified based on the decision of the subject expert.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Explain and verify the characteristics and operation of semiconductor devices, rectifiers, and transistor configurations through laboratory experiments.	BL1 & BL2
CO2	Apply electronic circuit concepts to construct and test oscillators, multivibrators, and Op-Amp-based applications.	BL3
CO3	Analyse amplifier biasing, frequency response, oscillation conditions, waveform transformations, and DAC characteristics using measured data and CRO observations.	BL4
CO4	Utilize AI/LLM tools along with Python/MATLAB simulations to interpret electronic circuit behaviour, estimate parameters, and	BL4

CO	Course Outcome Statement	Bloom's Level
	validate theoretical models.	
CO5	Design and evaluate AI-assisted predictive models, regression-based analysis systems, and simulation frameworks for electronic circuits and applications.	BL5

CO – PO Articulation Matrix

Note: 1 – Low, 2 – Medium, 3 – High, '-' – No Correlation

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	BL1 & BL2	3	2	-	2	2	-	-	1	-	-	1	3	2	2
CO2	BL3	3	3	3	2	3	-	-	1	1	-	1	3	2	2
CO3	BL4	2	3	3	3	3	1	-	1	1	-	2	3	2	2
CO4	BL4	2	3	2	3	3	1	1	1	1	-	3	3	2	2
CO5	BL5	2	2	3	3	3	1	1	2	2	1	3	3	2	2

II Year I Sem

Course Code	AI-ENABLED ELECTRICAL CIRCUIT ANALYSIS AND SIMULATION LAB	L	T	Pr	P	Credits
26AEE02301P	II Year I Sem	0	0	0	3	1.5
Pre-requisite	AI Assisted Electrical Circuit Analysis					

Course Objectives

- To perform experiments on fundamental electrical circuits to understand circuit laws, theorems, and network parameters.
- To analyze resonance, transient response, and two-port networks using experimental and computational techniques.
- To apply modern simulation and open-source software tools for circuit modelling and verification.
- To integrate Artificial Intelligence and data-driven methods for analyzing circuit behaviour and predicting responses.
- To develop teamwork, technical documentation, and experimental investigation skills.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the principles of resonance, network theorems, transient response, two-port network parameters, inductance, coupling coefficient, and power measurement in AC/DC electrical circuits.	L1 (Remember) L2 (Understand)
CO2	Apply experimental methods to verify network theorems, determine circuit parameters, analyze resonance conditions, measure active/reactive power, and evaluate transient responses in electrical networks.	L3 (Apply)
CO3	Analyze the behaviour of RLC circuits, locus diagrams, coupled circuits, balanced and unbalanced loads, and interconnected two-port networks using measured and simulated data.	L4 (Analyze)
CO4	Apply and analyze AI and machine learning techniques such as regression models, neural networks, symbolic AI tools, and LSTM models for prediction, theorem verification, transient analysis, and parameter estimation in electrical circuits.	L3 (Apply) L4 (Analyze)
CO5	Design AI-assisted experimental models and intelligent electrical circuit analysis systems for automated response prediction, theorem verification, resonance estimation, and two-port network characterization using simulation and measured data.	L4 (Analyze) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	L1, L2	3	2	1	2	1	–	–	–	–	–	1	3	1	1
CO2	L3	3	3	1	3	2	–	–	–	–	–	1	3	2	1
CO3	L4	2	2	2	2	3	–	–	1	1	–	2	2	2	2
CO4	L3, L4	2	3	2	2	3	1	–	1	–	–	3	2	3	3
CO5	L4, L6	1	2	1	2	2	–	1	3	3	1	2	1	2	2

List of Experiments

Any 10 of the following experiments are to be conducted:

1. Determination of self, mutual inductances, and coefficient of coupling
2. Series and parallel resonance
3. Locus diagrams of R-L (L Variable) and R-C (C Variable) series circuits
4. Verification of Superposition theorem using AC Excitation
5. Verification of Thevenin's and Norton's Theorems with DC Excitation
6. Verification of Maximum power transfer theorem with DC & AC Excitation
7. Verification of Compensation theorem with DC Excitation
8. Verification of Reciprocity Theorem and Millman's Theorem using DC & AC Excitation
9. Determination of Z and Y parameters
10. Determination of ABCD and Hybrid parameters
11. Series, Parallel and Cascaded connection of two port networks
12. Transient Response of RLC Series & Parallel circuits
13. Measurement of Active Power and Reactive Power for Balanced loads
14. Measurement of Active Power and Reactive Power for Unbalanced loads

AI Integration – Minimum two of the following experiments are to be conducted:

1. Use machine learning regression models to predict inductance and coupling coefficient from experimental voltage/current data.
2. AI-based estimation of resonant frequency and Q-factor.
3. AI-based prediction of circuit responses using neural networks.
4. Automatic theorem verification using symbolic AI tools.
5. AI-generated practice circuits and solution verification.
6. Machine learning models to estimate two-port parameters from measured data.
7. Time-series AI models (LSTM) to predict transient voltage/current.

References

1. Engineering Circuits Analysis, Jack Kemmerly, William Hayt and Steven Durbin, Tata McGraw Hill Education, 2005, Sixth Edition.
2. Network Analysis, M. E. Van Valkenburg, Pearson Education, 2019, Revised Third Edition.

Online Learning Resources / Virtual Labs & Open-Source Software

Open-Source Software:

- Python (NumPy, SciPy, Pandas)

- Scikit-learn for regression models
- LTspice for circuit simulation
- Jupyter Notebook

Open-Source Course Links (for AI + Circuits Learning):

- MIT OpenCourseWare – Circuits and Electronics: https://ocw.mit.edu
- Coursera – AI for Everyone / Machine Learning
- edX – Electrical Engineering and AI courses
- NPTEL – Electrical Circuits and Machine Learning
- Fast.ai – Practical Deep Learning

II Year I Sem

Course Code	INTELLIGENT CONTROL OF DC MACHINES AND TRANSFORMERS LAB	L	T	Pr	P	Credits
26AEE02303P	II Year I Sem	0	0	0	3	1.5
Pre-requisite	DC Machines and Transformers with Intelligent Control					

Course Objectives

- To perform experimental investigations on DC machines and transformers to understand their operating characteristics.
- To analyze experimental data for determining efficiency, losses and performance of electrical machines.
- To study different testing methods used for evaluating the performance of DC machines and transformers.
- To utilize computational tools and data visualization techniques for analyzing experimental results.
- To apply AI-based tools for intelligent monitoring, performance prediction and fault analysis of electrical machines.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the construction, operation, testing procedures, and characteristics of DC machines and transformers.	L1 (Remember) L2 (Understand)
CO2	Understand and apply various testing methods for determining efficiency, regulation, speed control, and performance characteristics of electrical machines.	L2 (Understand) L3 (Apply)
CO3	Apply experimental procedures and data acquisition techniques for analyzing machine and transformer performance.	L4 (Analyze)
CO4	Analyze performance curves, losses, efficiency, and operational behavior of electrical machines using conventional and AI-based methods.	L3 (Apply) L4 (Analyze)
CO5	Design models for performance prediction, maintenance, and visualization of electrical machine data using conventional and AI techniques.	L4 (Analyze) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	L1, L2	3	2	–	1	2	–	–	–	1	1	–	3	1	1
CO2	L2, L3	3	3	1	2	2	–	–	–	1	1	–	3	2	1
CO3	L4	3	3	2	2	3	–	–	–	1	1	–	3	2	2
CO4	L3, L4	3	3	2	3	3	–	1	–	1	1	–	2	2	2
CO5	L4, L6	3	3	3	3	3	1	1	–	2	1	1	2	3	3

List of Experiments

Any 10 of the following experiments are to be conducted:

1. Speed control of DC shunt motor by Field Current and Armature Voltage Control.
2. Brake test on DC shunt motor – Determination of performance curves.
3. Swinburne's test – Predetermination of efficiencies as DC Generator and Motor.
4. Hopkinson's test on DC shunt Machines.
5. Load test on DC compound generator – Determination of characteristics.
6. Load test on DC shunt generator – Determination of characteristics.
7. Fields test on DC series machines – Determination of efficiency.
8. Brake test on DC compound motor – Determination of performance curves.
9. OC & SC tests on single phase transformer.
10. Sumpner's test on single phase transformer.
11. Scott connection of transformers.
12. Parallel operation of Single-phase Transformers.
13. Separation of core losses of a single-phase transformer.

AI Applications in the Laboratory – Minimum two of the following experiments are to be conducted:

1. Machine learning models for predicting efficiency and performance of DC machines.
2. AI-based speed-torque curve prediction for motors.
3. Predictive analytics for DC machine performance and energy losses.
4. Neural network-based estimation of transformer losses and efficiency.
5. Data visualization of experimental results using AI tools.
6. AI-assisted load sharing of transformer.

References

• Virtual Lab (IITR): https://ems-iitr.vlabs.ac.in/List%20of%20experiments.html

Online Learning Resources / Virtual Labs & Open-Source Software

Open-Source Software for AI Applications:

- Python (NumPy, Pandas, Matplotlib)
- Scikit-Learn
- TensorFlow / PyTorch
- Jupyter Notebook

- GNU Octave / Scilab
- OpenModelica
- Elmer FEM

II Year - I Sem.

C PROGRAMMING AND DATA STRUCTURES

Course Code	Year/Sem	L	T	Pr	P	Credits

Course Code	Year/Sem	L	T	Pr	P	Credits
26ACS05301S		1	0	0	2	2

Course Objectives

- Understand the fundamentals of C programming language, data types, operators, control statements and arrays.
- Explain functions, recursion, pointers, storage classes, strings, structures, unions, and command line arguments in C programming.
- Apply stack and queue concepts for solving computational problems using arrays and linked representations.
- Analyze linked lists, trees, graphs, searching, and sorting techniques for efficient data organization and processing.
- Design and develop C programs using suitable data structures and algorithms for real-world engineering applications.

Course Content

UNIT I: Introduction to C Language

Core Content: C language elements, variable declarations and data types, operators and expressions, decision statements – If and switch statements, loop control statements – while, for, do-while statements, arrays.

UNIT II: Functions, Pointers and Storage Classes

Core Content: Functions: types of functions – Recursion and argument passing – pointers – storage allocation – pointers to functions – expressions involving pointers – Storage classes: auto, register, static, extern – Structures – Unions – Strings – string handling functions and Command line arguments.

Experiments:

1. Write C programs that use both recursive and non-recursive functions: (i) to find the factorial of a given integer; (ii) to find the GCD (greatest common divisor) of two given integers.
2. Write a C program to find both the largest and smallest number in a list of integers.
3. Write a C program that uses functions to insert a sub-string into a given main string from a given position.
4. Write a C program that uses functions to delete n characters from a given position in a given string.
5. Write a C program that displays the position or index in the string S where the string T begins, or –1 if S doesn't contain T.
6. Write a C program to count the lines, words and characters in a given text.
7. Write a C program to demonstrate the following parameter passing mechanisms: (i) call-by-value; (ii) call-by-reference.

UNIT III: Stacks and Queues

Core Content: Data Structures: Overview of data structures – stacks and queues – representation of a stack – stack related terms – operations on a stack – implementation of a stack – evaluation of arithmetic expressions – infix, prefix, and postfix notations – evaluation of postfix expression – conversion of expression from infix to postfix – recursion – queues – various positions of queue – representation of queue – insertion, deletion and searching operations.

UNIT IV: Linked Lists

Core Content: Singly linked list – dynamically linked stacks and queues – polynomials using singly linked lists – using circularly linked lists – insertion, deletion and searching operations – doubly linked lists and its operations – circular linked lists and its operations.

Experiments:

1. Write C programs that implement stack using arrays.
2. Write C programs that implement Queue using arrays.
3. Write a C program that uses Stack operations to evaluate postfix expression.
4. Write a C program that uses functions to perform the following operations on singly linked list: (i) Creation; (ii) Insertion; (iii) Deletion.
5. Write a C program that uses functions to create a binary tree of integers.
6. Write C programs for demonstration of linear search.
7. Write C programs for demonstration of binary search.
8. Write a C program for demonstration of bubble sort.
9. Write a C program for demonstration of selection sort.

UNIT V: Trees, Graphs, Searching and Sorting

Core Content: Trees: Tree terminology – representation; Binary trees – representation, binary tree traversals, binary tree operations; Graphs – graph terminology, graph representation, elementary graph operations – Breadth First Search (BFS) and Depth First Search (DFS) – connected components – spanning trees; Searching and Sorting: sequential search – binary search – exchange (bubble) sort – selection sort – insertion sort.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the fundamentals of C language elements, data types, operators, control statements, arrays, and basic programming constructs.	Remember-L1, Explain-L2
CO2	Understand and apply functions, recursion, pointers, storage classes, strings, structures, unions, and command line arguments in C programs.	Understand-L2, Apply-L3
CO3	Apply stack and queue operations to solve problems involving expression evaluation and data processing.	Apply-L3
CO4	Analyze linked lists, trees, and graph traversal techniques for efficient	Analyze-L4

CO	Course Outcome Statement	Bloom's Level
	implementation of dynamic data structures.	
CO5	Design and develop efficient searching and sorting programs using appropriate algorithms and data structures in C.	Design-L5

CO – PO Articulation Matrix

Note: 1 – Low, 2 – Medium, 3 – High, '-' – No Correlation

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	L1, L2	3	2	-	-	2	-	-	-	-	1	-	2	1	1
CO2	L2, L3	3	3	1	-	2	-	-	-	1	1	-	2	1	1
CO3	L3	3	3	2	1	3	-	-	-	1	-	-	2	1	1
CO4	L4	3	3	2	2	3	-	-	-	1	-	-	2	1	1
CO5	L5	3	3	3	2	3	-	-	-	1	1	1	2	1	1

Textbooks

1. The C Programming Language, Brian W. Kernighan and Dennis M. Ritchie, Second Edition, Prentice Hall Publication.
2. Fundamentals of Data Structures in C, Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Computer Science Press.
3. Programming in C and Data Structures, J.R. Hanly, Ashok N. Kamthane and A. Ananda Rao, Pearson Education.
4. Richard F. Gilberg & Behrouz A. Forouzan, "Data Structures: A Pseudocode Approach with C", Second Edition, CENGAGE Learning, 2011.

Reference Books

1. E. Balaguruswamy, "C and Data Structures", 4th Edition, Tata Mc Graw Hill.
2. A.K. Sharma, Computer Fundamentals and Programming in C, 2nd Edition, University Press.
3. M.T. Somashekara, "Problem Solving Using C", PHI, 2nd Edition, 2009.

Web Resources & NPTEL Links

- NPTEL Course (Preview): https://onlinecourses.nptel.ac.in/noc26_cs53/preview
- NPTEL Course: https://nptel.ac.in/courses/106106127
- Programiz – C Programming: https://www.programiz.com/c-programming

- Programiz – DSA: https://www.programiz.com/dsa
- TutorialsPoint – C Programming: https://www.tutorialspoint.com/cprogramming/
- TutorialsPoint – Data Structures & Algorithms: https://www.tutorialspoint.com/data_structures_algorithms/

II Year I Sem

Course Code	ENVIRONMENTAL SCIENCE	L	T	Pr	P	Credits
26AHS00301M	Common to All Branches	2	0	0	0	0

Prerequisites: No specific prerequisites; open to all engineering branches.

COURSE OBJECTIVES

- Create awareness about environmental issues, natural resource management and the need for sustainable development.
- Explain the structure and function of various ecosystems and the significance of biodiversity and its conservation.
- Describe causes, effects and control of various types of pollution and principles of solid waste management.
- Examine social, ethical and legislative issues related to environment and sustainable development in India.
- Understand human population dynamics, environmental health linkages and the role of IT and AI tools in environmental monitoring.

COURSE CONTENT

UNIT I: Multidisciplinary Nature of Environmental Studies and Natural Resources (9 Hrs)

Core Content: Multidisciplinary nature of environmental studies: definition, scope and importance; need for public awareness. Natural resources: renewable and non-renewable resources; natural resources and associated problems; forest resources – use and over-exploitation, deforestation, case studies, timber extraction, mining, dams and other effects on forest and tribal people; water resources – use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams – benefits and problems; mineral resources – use and exploitation, environmental effects of extracting and using mineral resources, case studies; food resources – world food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies; energy resources – use and exploitation of conventional and non-conventional energy sources, case studies.

AI Integration: AI in natural resource monitoring: remote sensing and satellite imagery analysis using Google Earth Engine (free cloud platform) for real-time deforestation detection and land use change mapping. Machine learning for groundwater level prediction from historical well data (Python/scikit-learn). AI-based precision agriculture – reducing fertilizer and pesticide use through crop health monitoring using drone imagery and computer vision (overview). ML forecasting of renewable energy output (solar irradiance, wind speed) for energy resource planning.

Free/Open-Source AI Tools: Google Earth Engine (free cloud GIS); QGIS (free, open-source); Python (scikit-learn, Matplotlib) on Google Colab (free); Global Forest Watch (free dashboard).

UNIT II: Ecosystems and Biodiversity and its Conservation (9 Hrs)

Core Content: Ecosystems: concept of an ecosystem; structure and function of an ecosystem; producers, consumers and decomposers; energy flow in the ecosystem; ecological succession; food chains, food webs and ecological pyramids; introduction, types, characteristic features, structure and function of forest, grassland, desert and aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries). Biodiversity and its conservation: introduction and definition – genetic, species and ecosystem diversity; bio-geographical classification of India; value of biodiversity – consumptive use, productive use, social, ethical, aesthetic and option values; biodiversity at global, national and local levels; India as a mega-diversity nation; hot-spots of biodiversity; threats to biodiversity – habitat loss, poaching of wildlife, man-wildlife conflicts; endangered and endemic species of India; conservation of biodiversity – in-situ and ex-situ conservation.

AI Integration: AI-powered species identification: applications such as iNaturalist (free) use image recognition to identify plants, animals and insects from photographs, enabling citizen science biodiversity surveys. ML for biodiversity hotspot mapping using satellite imagery and species distribution models (MaxEnt – free). Computer vision for automated wildlife camera trap image analysis – identifying and counting species without manual review. Google Earth Engine for ecosystem health monitoring: NDVI time-series analysis for vegetation cover change detection. AI-assisted assessment of in-situ vs. ex-situ conservation effectiveness.

Free/Open-Source AI Tools: iNaturalist (free app/web); MaxEnt (free species distribution model); Google Earth Engine (free); QGIS (free); Python (rasterio, geopandas) on Google Colab.

UNIT III: Environmental Pollution and Solid Waste Management (9 Hrs)

Core Content: Environmental pollution: definition, cause, effects and control measures of air pollution, water pollution, soil pollution, marine pollution, noise pollution, thermal pollution and nuclear hazards. Solid waste management: causes, effects and control measures of urban and industrial wastes; role of an individual in prevention of pollution; pollution case studies; disaster management – floods, earthquake, cyclone and landslides.

AI Integration: AI in pollution monitoring and disaster management: ML-based Air Quality Index (AQI) prediction from meteorological data and traffic density (Python/scikit-learn/TensorFlow – free). IoT sensor networks with AI-driven anomaly detection for real-time water quality monitoring. Computer vision for automated solid waste sorting and classification (overview of AI-enabled sorting robots). OpenAQ platform (free, open data) for accessing global air quality sensor data for ML model training. AI and GIS (QGIS) for disaster risk zone mapping – flood inundation modelling, earthquake vulnerability assessment and cyclone track prediction.

Free/Open-Source AI Tools: OpenAQ (free global AQ data API); QGIS (free); Python (scikit-learn, TensorFlow, Matplotlib) on Google Colab (free); CPCB data portal (free).

UNIT IV: Social Issues and the Environment (9 Hrs)

Core Content: Social issues and the environment: from unsustainable to sustainable development; urban problems related to energy; water conservation, rain water harvesting, watershed management; resettlement and rehabilitation of people – its problems and concerns, case studies; environmental ethics – issues and possible solutions; climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust, case studies; wasteland reclamation; consumerism and waste products; Environment Protection Act; Air (Prevention and Control of Pollution) Act; Water (Prevention and Control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act; issues involved in enforcement of environmental legislation; public awareness.

AI Integration: AI for climate change and sustainable development: global climate model data analysis using Python (xarray, Matplotlib) from freely available CMIP6 datasets – visualising temperature rise, precipitation change and sea level scenarios. ML for carbon footprint estimation from energy consumption and transport data (Python/scikit-learn). AI-based watershed management: QGIS + Python for delineation, runoff modelling and rainwater harvesting potential mapping. Overview of AI in environmental compliance monitoring – automated identification of violations using satellite data. Digital tools for public environmental awareness: social media sentiment analysis for pollution perception studies (Python/NLP).

Free/Open-Source AI Tools: Python (xarray, Matplotlib, scikit-learn) on Google Colab (free); CMIP6 climate data (free, ESGF portal); QGIS (free); openLCA (free, LCA software).

UNIT V: Human Population and the Environment (9 Hrs)

Core Content: Human population and the environment: population growth, variation among nations; population explosion; family welfare programmes; environment and human health; human rights; value education; HIV/AIDS; women and child welfare; role of Information Technology in environment and human health; case studies. Field work: visit to a local area to document environmental assets – river/forest/grassland/hill/mountain; visit to a local polluted site – urban/rural/industrial/agricultural; study of common plants, insects and birds – river, hill slopes, etc.

AI Integration: AI in population studies and public health: ML-based population growth forecasting from census data using Python (time-series regression with scikit-learn / LSTM with TensorFlow). AI in public health surveillance – disease outbreak prediction and mapping using spatiotemporal ML models (overview of WHO FluNet, IDSP data). Role of Information Technology and AI in environmental health: wearable sensors for personal air quality monitoring; smartphone-based water quality testing with ML image analysis. Citizen science apps for field work data collection: iNaturalist, eBird (free) for recording plants, birds and insects during field visits; OpenStreetMap for documenting environmental assets. GIS-based environmental health mapping using QGIS (free).

Free/Open-Source AI Tools: iNaturalist (free); eBird (free); QGIS (free); Python (scikit-learn, TensorFlow) on Google Colab (free); OpenStreetMap (free).

COURSE OUTCOMES (COS)

CO	Course Outcome Statement	Bloom's Level
CO1	Recall and explain the multidisciplinary nature of environmental studies, define key concepts of scope and public awareness, and describe the uses, over-exploitation and associated environmental problems of forest, water, mineral, food and energy resources with reference to Indian and global case studies.	L1–L2
CO2	Describe the structure and function of different ecosystem types, explain ecological succession, food chains and ecological pyramids, and outline the concept and classification of biodiversity, its value, threats and conservation strategies (in-situ and ex-situ) including India's status as a mega-diversity nation.	L2–L3
CO3	Identify causes, effects and control measures of air, water, soil, marine, noise, thermal and nuclear pollution; apply principles of solid waste management for urban and industrial wastes; analyze pollution case studies and evaluate disaster management strategies for floods, earthquakes, cyclones and landslides.	L3–L4
CO4	Analyze social issues related to unsustainable development; evaluate the impacts of climate change, global warming, acid rain and ozone layer depletion; examine Indian environmental legislation (EPA, Air Act, Water Act, Wildlife Protection Act, Forest Conservation Act) and assess their enforcement challenges.	L4–L5
CO5	Describe population growth trends and their environmental impacts; identify links between environment and human health; demonstrate the role of information technology and AI-based tools in environmental monitoring, public health surveillance and field data collection; apply digital tools (QGIS, Python, iNaturalist) for environmental documentation.	L2–L3

CO–PO / PSO ARTICULATION MATRIX

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	L1-L2	2	1	–	–	1	3	3	2	–	2	–	2	2	1
CO2	L2-L3	2	2	–	1	2	3	3	2	1	2	–	2	2	2

CO3	L3-L4	2	2	2	2	2	3	3	3	1	2	1	2	2	2
CO4	L4-L5	2	2	2	2	2	3	3	3	1	2	1	3	2	2
CO5	L2-L3	1	1	–	1	3	3	3	2	1	2	–	3	1	2

CO–PO Correlation Highlights: Environmental Science demonstrates its strongest correlation with PO6 (Engineer & Society), PO7 (Environment & Sustainability) and PO8 (Ethics) across all five COs, reflecting the direct societal and environmental focus of the course. PO12 (Lifelong Learning) is well addressed through all COs as the course creates lasting environmental awareness. CO4 and CO5 address PO5 (Modern Tools) through AI/GIS-based environmental monitoring tools. As an audit course, the PSO correlations are secondary, though CO3 and CO4 contribute to PSO1 through pollution engineering and environmental management content relevant to civil infrastructure projects.

TEXTBOOKS

1. Erach Bharucha, "Textbook of Environmental Studies for Undergraduate Courses", for University Grants Commission, Universities Press.

2. Palaniswamy, "Environmental Studies", Pearson Education.

3. S. Azeem Unnisa, "Environmental Studies", Academic Publishing Company.

4. K. Raghavan Nambiar, "Text Book of Environmental Studies for Undergraduate Courses as per UGC Model Syllabus", Scitech Publications (India) Pvt. Ltd.

REFERENCE BOOKS

1. Deeksha Dave and E. Sai Baba Reddy, "Textbook of Environmental Science", Cengage Publications.

2. M. Anji Reddy, "Text Book of Environmental Sciences and Technology", BS Publications.

3. J. P. Sharma, "Comprehensive Environmental Studies", Laxmi Publications.

4. J. Glynn Henry and Gary W. Heinke, "Environmental Sciences and Engineering", Prentice Hall of India Private Limited.

5. G. R. Chatwal, "A Text Book of Environmental Studies", Himalaya Publishing House.

6. Gilbert M. Masters and Wendell P. Ela, "Introduction to Environmental Engineering and Science", Prentice Hall of India Private Limited.

WEB RESOURCES & NPTEL / ONLINE LINKS

- NPTEL: Environmental Science – https://nptel.ac.in/courses/124107002
- Google Earth Engine (free cloud geospatial platform) – https://earthengine.google.com
- Global Forest Watch (free deforestation monitoring) – https://www.globalforestwatch.org
- OpenAQ (free global air quality open data) – https://openaq.org
- iNaturalist (free citizen science biodiversity app) – https://www.inaturalist.org
- QGIS (free, open-source GIS) – https://qgis.org

- India Meteorological Department (free data) – https://mausam.imd.gov.in
- Central Pollution Control Board (CPCB) data portal (free) – https://cpcb.nic.in

RECOMMENDED FREE & OPEN-SOURCE AI/COMPUTATIONAL TOOLS

- Google Earth Engine (free cloud platform) – real-time satellite data analysis for deforestation, land use and vegetation change detection
- QGIS (free, open-source GIS) – environmental mapping, watershed delineation, disaster risk zone mapping, biodiversity hotspot analysis
- Python (NumPy, Pandas, Matplotlib, scikit-learn, TensorFlow) on Google Colab (free) – AQI prediction, population growth ML models, climate data analysis
- iNaturalist (free, open-source citizen science) – species identification using image recognition during field work; community biodiversity database
- eBird (free, Cornell Lab) – bird species recording and mapping during field visits; global bird distribution data
- OpenAQ (free, open-source) – global real-time and historical air quality data for ML model training
- MaxEnt (free) – species distribution modelling for biodiversity hotspot prediction from occurrence data
- Global Forest Watch (free dashboard) – monitoring deforestation and forest cover change using satellite imagery
- CMIP6 climate datasets (free, ESGF portal) – global climate model data for Python-based climate change scenario analysis

II Year II Sem

Course Code	BIOLOGY FOR ENGINEERS	L	T	Pr	P	Credits
26AHS0304T	Common to All Branches	2	0	0	0	2

Prerequisites: No specific prerequisites; open to all engineering branches. This course introduces biological concepts through the lens of engineering and artificial intelligence, emphasizing biomolecules, human systems, bio-inspired designs, and modern bioengineering trends, with practical case studies and AI-enabled problem-solving.

COURSE OBJECTIVES

- To provide foundational biological knowledge with engineering relevance.
- To enable teachers to demonstrate AI-driven applications in biology and bioengineering.
- To encourage interdisciplinary teaching that connects biology, engineering, and computational tools.
- To prepare students for innovation in healthcare, materials, and bio-industries.

COURSE CONTENT

UNIT I: Foundations of Biology and AI

Core Content: Cell as the basic unit of life. Carbohydrates, proteins, lipids, enzymes, vitamins, hormones.

AI Integration: AI-based visualization of biomolecules (e.g., AlphaFold). Case studies on AI in drug discovery.

Free/Open-Source AI Tools: Protein folding prediction using AlphaFold; use of DeepChem to simulate enzyme-substrate binding; comparison of AI-predicted protein structures with experimental data.

UNIT II: Biomolecules in Engineering Applications

Core Content: Carbohydrates in bioplastics (PHA, PLA). Proteins in food engineering (whey protein, plant analogs).

AI Integration: AI-driven generative design for biomaterials. Lipids in biodiesel & detergents.

Free/Open-Source AI Tools: PLA bioplastic design using ANSYS; AI-based optimization of protein-rich diets; design of a cellulose-based water filter using AI simulation; model biodiesel yield prediction using MATLAB.

UNIT III: Human Systems and Bio-Designs

Core Content: Brain as CPU (EEG, robotic prosthetics). Eye as camera (optical corrections, lens materials).

AI Integration: Heart as pump (ECG, stents). Kidney as filtration (dialysis systems).

Free/Open-Source AI Tools: AI-based ECG anomaly detection; robotic arm design inspired by neural signals; use of AI imaging tools to detect cataracts; simulate stent design using ANSYS.

UNIT IV: Bio-Inspired Materials and Mechanisms

Core Content: Photosynthesis → photovoltaic cells. Bird flight → GPS and aircraft navigation.

AI Integration: Lotus leaf → superhydrophobic surfaces. Kingfisher beak → bullet train design.

Free/Open-Source AI Tools: Model solar energy conversion inspired by photosynthesis; AI simulation of lotus-leaf hydrophobicity; kingfisher-inspired aerodynamic modelling; use of AI to design self-cleaning surfaces.

UNIT V: Emerging Trends in Bioengineering

Core Content: Muscular and skeletal scaffolds. 3D bioprinting of tissues (ear, bone, skin).

AI Integration: AI in medical imaging (MRI, CT scans). AI in diagnostics and predictive healthcare.

Free/Open-Source AI Tools: AI-based tumor detection in MRI scans; 3D bioprinting case study of bone tissue; use of AI to classify medical images; design of a scaffold for bone regeneration using ANSYS.

COURSE OUTCOMES (COS)

CO	Course Outcome Statement	Bloom's Level
CO1	Explain the fundamentals of biology, bio molecules, and human systems relevant to engineering applications.	L1–L2
CO2	Apply AI tools and computational techniques for biological data	L3

	analysis, bio molecule visualization, and healthcare applications.	
CO3	Analyze bio molecules and bio-inspired systems for solving engineering problems and developing sustainable technologies.	L4
CO4	Design bio mimetic and bioengineering solutions using interdisciplinary engineering approaches and AI-enabled tools.	L5–L6
CO5	Evaluate emerging trends in bioengineering such as 3D bio printing, regenerative medicine, and AI-based diagnostics for societal applications.	L5

CO–PO ARTICULATION MATRIX

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

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	1	–	–	–	–	–	–	2
CO2	2	3	2	2	3	–	–	–	1	–	–	2
CO3	2	3	3	2	2	2	3	–	–	–	–	2
CO4	2	2	3	2	3	2	2	1	2	1	–	2
CO5	2	2	2	2	3	3	2	1	1	1	–	3

TEXTBOOKS

1. Biology for Engineers, Rajendra Singh C and Rathnakar Rao N, Rajendra Singh C and Rathnakar Rao N Publishing, Bengaluru, 2023.

2. Biology for Engineers, Thyagarajan S., Selvamurugan N., Rajesh M.P., Nazeer R.A., Thilagaraj W., Barathi S., and Jaganthan M.K., Tata McGraw-Hill, New Delhi, 2012.

REFERENCE BOOKS

1. Human Physiology, Stuart Fox, Krista Rompolski, McGraw-Hill eBook, 16th Edition, 2022.

2. Biology for Engineers, Arthur T. Johnson, CRC Press, Taylor and Francis, 2011.

3. Biomedical Instrumentation, Leslie Cromwell, Prentice Hall, 2011.

4. Biology for Engineers, Sohini Singh and Tanu Allen, Vayu Education of India, New Delhi, 2014.

5. Biomimetics: Nature-Based Innovation, Yoseph Bar-Cohen, 1st Edition, 2012, CRC Press.

6. Bio-Inspired Artificial Intelligence: Theories, Methods and Technologies, D. Floreano and C. Mattiussi, MIT Press, 2008.

7. 3D Bioprinting: Fundamentals, Principles and Applications, Ibrahim Ozbolat, Academic Press, 2016.

WEB RESOURCES & NPTEL / ONLINE LINKS

- NPTEL: Biology for Engineers – https://nptel.ac.in/courses/121106008
- Free Video Lectures (NPTEL) – https://freevideolectures.com/course/4877/nptel-biology-engineers-other-non-biologists
- MIT OpenCourseWare – Introduction to Biological Engineering Design – https://ocw.mit.edu/courses/20-020-introduction-to-biological-engineering-design-spring-2009
- MIT OpenCourseWare – Introduction to Bioengineering – https://ocw.mit.edu/courses/20-010j-introduction-to-bioengineering-be-010j-spring-2006
- Coursera – Biology courses – https://www.coursera.org/courses?query=biology
- NPTEL Online – https://onlinecourses.nptel.ac.in/noc19_ge31/preview
- Class Central – Biology – https://www.classcentral.com/subject/biology
- FutureLearn – Biology: Basic Concepts – https://www.futurelearn.com/courses/biology-basic-concepts

RECOMMENDED FREE & OPEN-SOURCE AI/COMPUTATIONAL TOOLS

- DeepChem – drug discovery simulations
- AlphaFold – protein structure prediction
- Generative Design AI – biomaterial innovation
- ANSYS – biomimetic engineering simulations
- MATLAB Simulink – biological system modeling
- AI Medical Imaging Tools – diagnostics and healthcare

II Year II Sem

Course Code	SUSTAINABILITY AND GREEN ENGINEERING	L	T	Pr	P	Credits
26AME3401T	II Year II Sem	2	0	0	0	2

Prerequisites: No specific prerequisites; open to all engineering branches.

COURSE OBJECTIVES

- Understand the concepts of sustainability, carbon cycle, and the environmental impact of civil engineering activities.
- Evaluate sustainable construction materials, their life cycle, and environmental performance.
- Apply energy calculations for buildings including embodied energy, operational energy, and life cycle energy.
- Assess green building standards, energy codes (ECSBC), and performance rating systems (LEED, GRIHA).
- Integrate AI/ML tools for carbon footprint analysis, energy optimization, and sustainable design decision-making.

COURSE CONTENT

UNIT I: Fundamentals of Sustainability & Carbon Cycle

Core Content: Definition of sustainability: three pillars (economic, social, environmental). Brundtland definition. Sustainable Development Goals (SDGs): SDGs 6, 7, 9, 11, 12, 13. Carbon cycle: biotic and abiotic components. Greenhouse gases: CO_2, CH_4, N_2O, HFCs. Carbon dioxide equivalent (CO_2e). CO_2 contribution from cement production (~8% global emissions), steel, glass, aluminium. Carbon footprint: definition, calculation methodology, carbon accounting. Climate change: global warming, sea level rise, extreme events.

AI Integration: AI for carbon footprint tracking: overview of ML-based life cycle assessment tools. Python-based carbon footprint calculator: computing embodied carbon from material quantities. Introduction to climate modelling data processing using Python (pandas, matplotlib). Online tools: Global Carbon Atlas, openLCA (free) for life cycle assessment.

Free/Open-Source AI Tools: openLCA (free, open-source LCA software); Python (Pandas, Matplotlib) on Google Colab; Global Carbon Project tools; SimaPro (student trial).

UNIT II: Sustainable Construction Materials

Core Content: Construction materials and indoor air quality: VOCs, formaldehyde. No/low cement concrete: supplementary cementitious materials (fly ash, GGBS, silica fume, rice husk ash) – properties, mix proportions. Recycled aggregate concrete: sources, properties, durability. Alternative binders: alkali-activated materials (AAM), geopolymer concrete. Manufactured sand (M-sand): processing, quality. Bamboo, timber, and fibre-reinforced composites as sustainable alternatives. Durability and service life assessment. Life cycle of construction materials.

AI Integration: ML for sustainable material selection: multi-criteria decision analysis using Python. AI-based prediction of concrete strength from mix proportions using regression models (scikit-learn). Computer vision for recycled aggregate quality assessment. Introduction to materials informatics: using ML to discover new sustainable binders. Comparative LCA using openLCA.

Free/Open-Source AI Tools: scikit-learn (free Python ML library); openLCA (free); Python (NumPy, scikit-learn) on Google Colab; Concrete Mixture Proportioner (free online NRMCA tool).

UNIT III: Embodied Energy & Operational Energy in Buildings

Core Content: Definition of embodied energy: extraction, processing, manufacturing, transport, construction, demolition. Operational energy: heating, cooling, lighting, hot water. Life cycle energy: sum of embodied and operational energy. Calculation of embodied energy for common construction materials (energy intensity values). Energy concept: primary energy, renewable vs. non-renewable energy. Embodied energy calculation for a simple building structure. Energy balance: reducing operational energy vs. increasing insulation (embodied energy trade-off).

AI Integration: Python-based embodied energy calculator for building elements. AI for building energy optimization: overview of EnergyPlus (free) and OpenStudio (free) simulations. ML for operational energy prediction from building parameters (Linear Regression, Random Forest using scikit-learn). Digital twin for energy monitoring: concept and tools. Building energy benchmarking using ML: comparison with national average.

Free/Open-Source AI Tools: EnergyPlus (free, US DOE simulation tool); OpenStudio (free); Python (scikit-learn, NumPy) on Google Colab; openLCA (free).

UNIT IV: Green Building Rating Systems & Energy Codes

Core Content: Energy Conservation and Sustainability Building Code (ECSBC-2023): scope, mandatory and prescriptive requirements, OTTV calculations, compliance path. Green building rating systems: LEED (Leadership in Energy and Environmental Design) – credit categories, certification levels; GRIHA (Green Rating for Integrated Habitat Assessment) – India's national system, parameters; IGBC rating systems; BEE star rating for commercial buildings. Passive design strategies: building orientation, natural ventilation, daylighting, cool roofs. Net Zero Energy Buildings (NZEB): definition, pathways.

AI Integration: AI for automated LEED/GRIHA compliance checking: overview of existing tools. ML for building performance prediction: predicting LEED score from building design parameters. BIM-based energy analysis: linking Revit model to energy simulation. Smart building systems: AI-driven HVAC optimization. IoT sensors for real-time energy monitoring. Case study: AI-optimized green building design.

Free/Open-Source AI Tools: EnergyPlus (free); OpenStudio (free); Autodesk Revit student version with Energy Analysis add-in (free); Python for data analysis (Google Colab).

UNIT V: Circular Economy, Waste Management & Environmental Sustainability

Core Content: Circular economy principles: reduce, reuse, recycle, recover, redesign. Waste hierarchy. Construction and demolition (C&D) waste: types, quantities, recycling potential. C&D waste management rules 2016. Deconstruction vs. demolition. Industrial waste in construction: fly ash, blast furnace slag, quarry dust, plastic waste utilization. Water recycling in construction. Green procurement policies. Carbon sequestration potential of green infrastructure: urban trees, green roofs, wetlands. Acid rain: causes, effects, prevention.

AI Integration: AI for construction waste prediction: ML models for waste generation estimation. Computer vision for C&D waste sorting (overview of automated sorting robots). Digital material passport: BIM-based end-of-life material tracking. ML for demolition waste characterization. Python for waste audit data analysis. Sustainable infrastructure planning using AI and GIS: overview of green infrastructure planning tools.

Free/Open-Source AI Tools: Python (scikit-learn, Pandas) on Google Colab; QGIS (free GIS for green infrastructure planning); openLCA (free LCA); i-Tree (free urban tree carbon quantification tool by USDA).

COURSE OUTCOMES (COS)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the principles of sustainability, SDGs, the carbon cycle, greenhouse gas accounting, and the environmental impact of civil engineering materials and activities, with focus on the cement and steel industries.	L1–L2
CO2	Apply knowledge of sustainable construction materials including supplementary cementitious materials, recycled aggregates, geopolymer concrete, and alternative binders to evaluate their environmental performance using life cycle assessment methods.	L3
CO3	Analyze the embodied energy and operational energy of building systems, perform energy calculations for building components, and compare the environmental performance of design alternatives using computational tools.	L4
CO4	Evaluate green building rating systems (LEED, GRIHA, ECBC), assess compliance of building designs with energy codes, and compare the sustainability performance of different building design strategies using simulation and AI tools.	L4–L5

CO5	Design sustainable civil engineering solutions integrating circular economy principles, C&D waste management, green infrastructure, and AI-driven optimization tools for minimizing carbon footprint and maximizing resource efficiency.	L5–L6

CO–PO / PSO ARTICULATION MATRIX

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	L1,L2	2	1	–	–	–	3	3	2	–	2	–	2	2	2
CO2	L3	2	2	2	2	2	2	3	2	1	2	–	2	2	2
CO3	L4	2	2	2	2	3	2	3	1	1	2	1	2	2	3
CO4	L4,L5	2	2	3	2	3	3	3	2	1	2	1	3	2	3
CO5	L5,L6	2	3	3	3	3	3	3	2	2	3	2	3	2	3

TEXTBOOKS

1. Kibert C.J., Sustainable Construction: Green Building Design & Delivery, Wiley, 4th Edition, 2016.

2. Goodhew S., Sustainable Construction Processes, Wiley-Blackwell, 2016.

REFERENCE BOOKS

1. Langston C.A. and Ding G.K.C., Sustainable Practices in the Built Environment, Butterworth Heinemann, 2011.

2. Spence W.P., Construction Materials, Methods and Techniques, Yesdee Publication, 3rd Edition, 2012.

3. ECSBC 2024 (Energy Conservation and Sustainability Building Code), Bureau of Energy Efficiency, Government of India.

4. National Building Construction Standards, 2026, Bureau of Indian Standards.

5. Goswami D., Kreith F. and Kreider J., Principles of Solar Engineering, Taylor & Francis, 2000.

WEB RESOURCES & NPTEL / ONLINE LINKS

- NPTEL: Sustainable Development – https://nptel.ac.in/courses/105105157
- NPTEL: Green Buildings – https://archive.nptel.ac.in/courses/105/102/105102195
- Bureau of Energy Efficiency (BEE), India – https://beeindia.gov.in
- IGBC (Indian Green Building Council) – https://igbc.in
- openLCA (free LCA software) – https://www.openlca.org

- EnergyPlus (free building energy simulation) – https://energyplus.net
- OpenStudio (free building energy modelling) – https://openstudio.net
- Global Carbon Project – https://www.globalcarbonproject.org

RECOMMENDED FREE & OPEN-SOURCE AI/COMPUTATIONAL TOOLS

- openLCA (free, open-source) – Life Cycle Assessment software for embodied carbon calculation
- EnergyPlus (free, US DOE) – Building energy simulation for operational energy analysis
- OpenStudio (free, NREL) – Building energy modeling interface for EnergyPlus
- Python (scikit-learn, Pandas, NumPy) on Google Colab – ML for energy prediction and material selection
- QGIS (free) – Green infrastructure planning and spatial sustainability analysis
- i-Tree Tools (free, USDA) – Urban tree carbon sequestration quantification
- SimaPro (student trial) – Life cycle assessment platform

II Year - II Sem.

Course Code	AI ASSISTED CONTROL SYSTEMS	L	T	Pr	P	Credits
26AEE02401T	II Year II Sem	2	0	2	0	3
Prerequisites:	Linear Algebra and Vector Calculus, Differential Equations & Vector Calculus, Electrical Circuits, AI Tools & Applications					

Course Objectives

- Understand classical control system concepts and mathematical modelling.
- Analyze time-domain and frequency-domain responses of systems.
- Evaluate system stability using analytical and graphical methods.
- Design controllers and compensators using classical and AI techniques.
- Apply state-space methods and AI tools for modern control problems.

Course Content

UNIT I: Control Systems Concepts

Open loop and closed loop control systems and their differences – Examples of control systems – Classification of control systems, Feedback characteristics, Effects of positive and negative feedback, Mathematical models – Differential equations of translational and rotational mechanical systems and electrical systems, Analogous Systems, Block diagram reduction methods – Signal flow graphs – Reduction using Mason's gain formula. Principle of operation of DC and AC Servo motor, Transfer function of DC servo motor – AC servo motor, Synchros.

AI Integration:AI-assisted block diagram reduction and signal flow analysis, AI-based parameter estimation of transfer functions, Neural network modelling of servo motor behaviour, Visualization of analogous systems and transfer functions

Free/Open-Source AI Tools:

- Python (Pandas, Matplotlib)
- OpenModelica
- TensorFlow / PyTorch (basic modeling)

UNIT II: Time Response Analysis

Standard Test Signals – Step Response – Impulse Response – Time response of first order systems – Characteristic Equation of Feedback control systems, Transient response of second order systems – Time domain specifications – Steady state response – Steady state errors and error constants, P, PI, PID Controllers.

AI Integration: AI-based prediction of transient and steady-state response, Visualization of first and second order system responses, Machine learning analysis of time-domain specifications, Intelligent tuning of P, PI, and PID controllers

Free/Open-Source AI Tools:

- Python (control systems library)
- MATLAB alternative: Scilab Xcos
- Scikit-learn, TensorFlow

UNIT III: Stability Analysis in Time Domain

The concept of stability – Routh's stability criterion – Stability and conditional stability – Limitations of Routh's stability. The Root locus concept – Construction of root loci – Effects of adding poles and zeros to G(s)H(s) on the root loci.

AI Integration: Classification of stable/unstable systems using ML, AI-assisted stability prediction using Routh criterion data, Machine learning analysis of pole-zero movement

Free/Open-Source AI Tools:

- Python (scikit-learn)
- Jupyter Notebook
- TensorFlow

UNIT IV: Frequency Response Analysis

Introduction, Frequency domain specifications – Bode diagrams – Determination of Frequency domain specifications and transfer function from the Bode Diagram – Stability Analysis from Bode Plots. Polar Plots – Nyquist Plots – Phase margin and Gain margin – Stability Analysis. Compensation techniques – Lag, Lead, Lag-Lead Compensator design in frequency domain.

AI Integration: AI-assisted generation and analysis of Bode plots, Machine learning prediction of gain margin and phase margin, Visualization of Nyquist and Polar plots, Intelligent compensator parameter optimization

Free/Open-Source AI Tools:

- Python (Matplotlib), Scikit-learn
- Jupyter Notebook
- TensorFlow

UNIT V: State Space Analysis of Continuous Systems

Concepts of state, state variables and state model – State models: differential equations & Transfer function models – Block diagrams. Diagonalization, Transfer function from state model, Solving the Time invariant state Equations – State Transition Matrix and its Properties. System response through State Space models. The concepts of controllability and observability, Duality between controllability and observability.

AI Integration: AI-assisted state-space modeling and system response prediction, Visualization of state trajectories and transition matrices, Machine learning analysis of controllability and observability, Comparative analysis of transfer function and state-space models using AI

Free/Open-Source AI Tools:

- Python (NumPy, SciPy)
- OpenModelica
- TensorFlow, Scikit-learn

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the fundamental concepts of control systems, feedback mechanisms, mathematical modelling, and servo systems with and without AI-assisted visualization tools.	L1 (Remember) L2 (Understand)
CO2	Understand the time response characteristics, steady-state analysis,	L2

	and controller actions of control systems using conventional and AI-based simulation techniques.	(Understand)
CO3	Apply stability analysis methods such as Routh criterion and Root Locus techniques for evaluating control system performance using analytical and AI-assisted tools.	L3 (Apply)
CO4	Analyze frequency response characteristics, compensation techniques, and stability margins using classical and AI-enabled computational methods.	L4 (Analyze)
CO5	Design state-space models, controllability and observability analysis, and intelligent control solutions using conventional and AI-based approaches.	L4 (Analyze) L5 (Evaluate)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**L1, L2**	3	2	–	–	2	–	–	–	–	1	–	3	1	1
CO2	**L2**	3	3	1	1	3	–	–	–	1	1	–	3	2	1
CO3	**L3**	3	3	2	2	3	–	–	–	1	1	–	3	2	2
CO4	**L4**	3	3	3	3	3	–	1	–	1	1	–	3	3	3
CO5	**L4, L5**	3	3	3	3	3	1	1	–	2	2	1	3	3	3

Textbooks

1. Modern Control Engineering by Katsuhiko Ogata, Prentice Hall of India Pvt. Ltd., 5th edition, 2010.
2. Control Systems Engineering by I. J. Nagrath and M. Gopal, New Age International (P) Limited Publishers, 5th edition, 2007.

Reference Books

1. Control Systems Principles & Design by M. Gopal, 4th Edition, McGraw Hill Education, 2012.
2. Automatic Control Systems by B. C. Kuo and Farid Golnaraghi, John Wiley and Sons, 8th edition, 2003.
3. Feedback and Control Systems, Joseph J. DiStefano III, Allen R. Stubberud & Ivan J. Williams, 2nd Edition, Schaum's Outlines, McGraw Hill Education, 2013.
4. Control System Design by Graham C. Goodwin, Stefan F. Graebe and Mario E. Salgado, Pearson, 2000.
5. Feedback Control of Dynamic Systems by Gene F. Franklin, J.D. Powell and Abbas Emami-Naeini, 6th Edition, Pearson, 2010.

Web Resources & NPTEL Links

- NPTEL – Control Systems: https://nptel.ac.in/courses/108102043
- NPTEL – Advanced Control Systems: https://nptel.ac.in/courses/108106098

II Year II Sem

Course Code	AI-ASSISTED CONTROL OF INDUCTION AND SYNCHRONOUS MACHINES	L	T	Pr	P	Credits
26AEE02403T	II Year II Sem	2	0	2	0	3
Prerequisites:	Electrical Circuits & DC Machines, AI Tools & Applications					

Course Objectives

- Understand the construction, working principles, and performance characteristics of induction motors and synchronous machines.
- Analyze torque-slip characteristics, equivalent circuits, and testing methods for induction motors.
- Evaluate voltage regulation, synchronization methods, and performance of synchronous generators.
- Apply AI/ML tools for prediction, monitoring, and control of induction and synchronous machines.
- Design intelligent drive systems and AI-assisted monitoring frameworks for rotating electrical machines.

Course Content

UNIT I: 3-Phase Induction Motors

Construction of Squirrel cage and Slipring induction motors – Production of rotating magnetic field – Principle of operation – Rotor emf and rotor frequency – Rotor current and power factor at standstill and during running conditions – Rotor power input, rotor copper loss and mechanical power developed and their inter-relationship – Equivalent circuit – Phasor diagram, Applications.

AI Integration: Machine learning models for predicting slip, efficiency and torque from motor operating data, AI-assisted analysis of equivalent circuits and phasor diagrams, Intelligent monitoring of induction motor operating conditions, Digital twin modelling of induction motors

Free/Open-Source AI Tools:

- Python, Scikit-learn
- Jupyter Notebook
- GNU Octave

UNIT II: Performance of 3-Phase Induction Motors

Torque equation – Expressions for maximum torque and starting torque – Torque-slip characteristics – Double cage and deep bar rotors – No load, Brake test and Blocked rotor tests – Circle diagram for predetermination of performance – Methods of starting – Starting current and torque calculations – Speed control of induction motor with V/f control method, rotor resistance control and rotor emf injection technique – Crawling and cogging – Induction generator operation.

AI Integration: Neural networks for torque-speed curve prediction, AI-assisted speed control analysis using V/f control, AI-based visualization of circle diagram characteristics, Intelligent analysis of starting torque and starting current

Free/Open-Source AI Tools:

- TensorFlow
- PyTorch
- Scilab

UNIT III: Synchronous Generator

Constructional features of non-salient and salient pole type alternators – Armature windings – Distributed and concentrated windings – Distribution & pitch factors – EMF equation – Armature reaction – Voltage regulation by synchronous impedance method – MMF method and Potier triangle method – Two reaction analysis of salient pole machines – Methods of synchronization – Slip test – Parallel operation of alternators.

AI Integration: AI-based voltage regulation prediction using load data, Smart monitoring and performance analysis of alternators, Machine learning prediction of generator efficiency and performance, AI-assisted synchronization analysis of alternators

Free/Open-Source AI Tools:

- OpenModelica
- Python
- Ngspice

UNIT IV: Synchronous Motor

Synchronous motor principle and theory of operation – Effect of excitation on current and power factor – Synchronous condenser – Expression for power developed – Hunting and its suppression – Methods of starting, Applications.

AI Integration: AI-based optimization for power factor improvement, Reinforcement learning for voltage and excitation control, AI-assisted prediction of synchronous motor performance, Intelligent monitoring of hunting and stability conditions

Free/Open-Source AI Tools:

- Python (Keras, SciPy)
- TensorFlow
- PyTorch

UNIT V: Special Machines

Single phase induction motors – Constructional features – Equivalent circuit – Starting methods; Principle and operation of Shaded pole motor. Principle of operation: Reluctance motor – Stepper motor – BLDC motor and universal motor – Applications.

AI Integration:AI-based sensorless control of BLDC motors, Reinforcement learning for stepper motor positioning control, AI-assisted performance analysis of single-phase induction motors, Data-driven analysis of universal motor performance

Free/Open-Source AI Tools:

- Scilab Xcos
- Keras
- Python

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the construction, operating principles, characteristics, and applications of induction motors,	L1 (Remember) L2 (Understand)

	synchronous machines, and special electrical machines with AI-assisted visualization tools.	
CO2	Understand the performance characteristics, testing methods, starting techniques, speed control methods, and synchronization of electrical machines using conventional and AI-based approaches.	L2 (Understand)
CO3	Apply equivalent circuits, torque equations, performance analysis methods, and AI-assisted computational tools for evaluating machine behavior.	L3 (Apply)
CO4	Analyze voltage regulation, stability, power factor improvement, speed control, and machine characteristics using analytical and AI-enabled techniques.	L4 (Analyze)
CO5	Design intelligent models, smart drive systems, and AI-assisted monitoring and control techniques for electrical machines using modern open-source platforms.	L5 (Evaluate) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**L1, L2**	3	2	–	–	2	–	–	–	–	–	–	3	1	1
CO2	**L2**	3	3	1	1	3	–	–	–	1	–	–	3	2	1
CO3	**L3**	3	2	2	2	3	–	–	–	1	–	–	3	2	1
CO4	**L4**	2	2	3	3	3	–	1	–	1	–	–	2	3	2
CO5	**L5, L6**	2	3	3	3	3	1	1	–	2	–	1	2	3	3

Textbooks

1. Electrical Machinery, Dr. P.S. Bhimbra, Khanna Publishing, 2021, First Edition.
2. Performance and Analysis of AC Machines by M.G. Say, CBS, 2002.

Reference Books

1. Electrical Machines, D.P. Kothari and I.J. Nagrath, McGraw Hill Education, 2017, Fifth Edition.
2. Theory & Performance of Electrical Machines by J.B. Gupta, S.K. Kataria & Sons, 2007.
3. Electric Machinery, A.E. Fitzgerald, Charles Kingsley, Stephen D. Umans, McGraw-Hill, 2020, Seventh edition.

Web Resources & NPTEL Links

- NPTEL – Electrical Machines: https://nptel.ac.in/courses/108/105/108105131
- NPTEL – Power Electronics: https://nptel.ac.in/courses/108106072

II Year II Sem

Course Code	INTELLIGENT POWER SYSTEMS – I	L	T	Pr	P	Credits
26AEE02404T	II Year II Sem	2	0	2	0	3
Prerequisites:	Electrical Circuits, Electrical Machines, AI Tools & Applications					

Course Objectives

- Understand the working principles and components of various power generation plants.
- Analyze the structure and operation of substations and distribution systems.
- Study underground cables and their electrical characteristics.
- Evaluate economic aspects and tariff structures of power systems.
- Apply AI-based computational tools for monitoring, prediction and optimization in modern power systems.

Course Content

UNIT I: Hydroelectric and Thermal Power Stations

Hydroelectric Power Stations: Selection of site – General layout of a hydroelectric power plant with brief description of major components and principle of operation. Thermal Power Stations: Selection of site – General layout of a thermal power plant – Brief description of components: boilers, super heaters, economizers and electrostatic precipitators, steam turbines (impulse and reaction), condensers, feed water circuit, cooling towers and chimney.

AI Integration: Predicting hydroelectric power generation using rainfall and reservoir data, AI-based thermal plant efficiency monitoring, Predictive maintenance of turbines and boilers using sensor data, Energy generation forecasting using regression models

Free/Open-Source AI Tools:

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

UNIT II: Nuclear Power Station

Location of nuclear power plant – Working principle – Nuclear fission – Nuclear fuels – Nuclear chain reaction – Nuclear reactor components: moderators, control rods, reflectors and coolants – Types of nuclear reactors and brief description of PWR, BWR and FBR. Radiation: Radiation hazards and shielding – Nuclear waste disposal.

AI Integration: AI-based reactor safety monitoring, Machine learning for radiation leakage detection, AI-based nuclear waste monitoring and classification, Neural network models for reactor stability prediction

Free/Open-Source AI Tools:

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

UNIT III: Substations

Air Insulated Substations – Indoor & outdoor substations – Substations layouts of 33/11 kV showing the location of all the substation equipment – Bus bar arrangements: simple arrangements like single bus bar, sectionalized single bus bar, double bus bar with one and two circuit breakers, main and transfer bus bar system. Gas Insulated Substations (GIS) – Advantages of GIS, constructional aspects, comparison of air insulated substations and GIS.

AI Integration: AI-based fault diagnosis in substations, Predictive maintenance of circuit breakers and transformers, AI-based load forecasting at substations, Smart grid-based substation automation

Free/Open-Source AI Tools:

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

UNIT IV: Distribution Systems and Underground Cables

Distribution Systems: Classification of Distribution systems, AC Distribution, Overhead versus Underground system – Connection schemes of Distribution system – Requirements of Distribution system – Design considerations in Distribution system. Underground Cables: Types of cables – Construction, types of insulating materials – Calculation of insulation resistance – Stress in insulation and power factor of cable – Capacitance of single and 3-Core belted cables. Grading of cables: capacitance grading and intersheath grading.

AI Integration: AI-based distribution load forecasting, Machine learning for cable insulation failure prediction, Smart grid monitoring of distribution networks, AI-based optimal routing for underground cable networks

Free/Open-Source AI Tools:

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

UNIT V: Economic Aspects & Tariff

Economic Aspects – Load curve – Load duration and integrated load duration curves – Discussion on economic aspects: connected load, maximum demand, demand factor, base and peak load plants, load factor, diversity factor, plant capacity factor and plant use factor. Tariff Methods – Costs of generation and their division into fixed, semi-fixed and running costs – Desirable characteristics of a tariff method – Tariff methods: simple rate, flat rate, block-rate, two-part, three-part and power factor tariff methods – Time of Day (ToD) tariff and Time of Use (ToU) tariff.

AI Integration: AI-based electricity demand forecasting, Machine learning for dynamic tariff optimization, AI-based consumer load pattern analysis, Optimization of electricity pricing strategies

Free/Open-Source AI Tools:

- OpenDSS
- MATPOWER (Octave)
- GridLAB-D
- Power Factory Educational Tools

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the principles, construction, operation, and components of hydroelectric, thermal, and nuclear power stations, substations, distribution systems, underground cables, and tariff structures in power systems.	L1 (Remember) L2 (Understand)
CO2	Apply power system concepts to analyze power plant operation, substation configurations, cable parameters, distribution schemes, load curves, and tariff calculations for electrical power engineering applications.	L3 (Apply)
CO3	Analyze the performance, efficiency, reliability, economic operation, fault conditions, and load characteristics of power plants, substations, distribution networks, and underground cable systems.	L4 (Analyze)
CO4	Apply and analyze power generation forecasting, fault diagnosis, predictive maintenance, smart grid monitoring, load forecasting, tariff optimization, and equipment health assessment in electrical power systems using AI techniques.	L3 (Apply) L4 (Analyze)
CO5	Design intelligent and energy-efficient power system solutions including smart substations, optimized distribution networks, dynamic tariff models, and AI-assisted monitoring systems using open-source simulation and data analytics tools.	L5 (Evaluate) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**L1, L2**	3	2	1	1	1	1	–	–	1	–	1	3	1	1
CO2	**L3**	3	3	2	2	2	2	–	–	1	–	2	3	2	1
CO3	**L4**	3	3	2	3	2	1	–	–	1	–	2	3	2	1
CO4	**L3, L4**	3	2	2	2	2	2	–	–	1	3	2	2	2	2
CO5	**L5, L6**	3	2	3	2	3	2	–	–	2	2	3	2	3	3

Textbooks

1. S. N. Singh, Electric Power Generation, Transmission and Distribution, PHI Learning Pvt Ltd, New Delhi, 2nd Edition, 2010.

2. J. B. Gupta, Transmission and Distribution of Electrical Power, S. K. Kataria and Sons, 10th Edition, 2012.

Reference Books

1. I. J. Nagarath & D.P. Kothari, Power System Engineering, McGraw-Hill Education, 3rd Edition, 2019.
2. C. L. Wadhwa, Generation, Distribution and Utilization of Electrical Energy, New Age International Publishers, 6th Edition, 2018.
3. V. K. Mehta and Rohit Mehta, Principles of Power System, S. Chand, 4th Edition, 2005.
4. Turan Gonen, Electric Power Distribution System Engineering, McGraw-Hill, 1985.
5. Handbook of Switchgear, BHEL, McGraw-Hill Education, 2007.

Web Resources & NPTEL Links

- NPTEL – Power Plant Engineering: https://nptel.ac.in/courses/108105058
- NPTEL – Power Distribution Engineering: https://nptel.ac.in/courses/108101039
- MIT OpenCourseWare – Electric Power Systems: https://ocw.mit.edu/courses/electric-power-systems
- Smart Grid Course: https://www.edx.org/learn/smart-grid
- Machine Learning for Engineers: https://scikit-learn.org/stable/tutorial

II Year II Sem

Course Code	AI INTEGRATED SIGNAL PROCESSING AND SYSTEMS	L	T	Pr	P	Credits
26AEE02405T	II Year II Sem	2	0	2	0	3
Prerequisites:	AI Tools & Applications					

Course Objectives

- Understand the fundamental concepts of signals, systems, and Fourier series representations.
- Analyze Fourier transform techniques, sampling theory, and signal reconstruction methods.
- Apply Laplace Transform, Z-Transform, DFT, and FFT for signal and system analysis.
- Evaluate digital filter design methods including FIR and IIR filters.
- Apply AI/ML tools for intelligent signal processing, spectral analysis, and filter design optimization.

Course Content

UNIT I: Fundamentals of Signals and Systems

Classification of Signals and Systems – Continuous-Time and Discrete-Time Signals – Basic Signal Operations: Time Shifting, Scaling, Folding – Elementary Signals and Sequences – System Properties: Linearity, Time Invariance, Causality, Stability, Memoryless Systems – Convolution and Correlation of Signals – Linear Time Invariant (LTI) Systems – Orthogonality of Signals – Fourier Series Representation of Periodic Signals – Trigonometric and Exponential Fourier Series – Properties of Fourier Series.

AI Integration: AI-based signal classification using Machine Learning, Machine learning-based identification of system properties, AI-assisted convolution and correlation analysis, Data-driven analysis of periodic and non-periodic signals

Free/Open-Source AI Tools:

- GNU Octave
- Python (NumPy, Pandas)
- Scikit-learn

UNIT II: Fourier Transform Techniques and Sampling Theory

Continuous-Time Fourier Transform (CTFT) – Properties of Fourier Transform – Frequency Spectrum Analysis – Discrete-Time Fourier Transform (DTFT) – Properties of DTFT – Energy Density Spectrum and Power Spectral Density – Sampling Theorem and Reconstruction of Signals – Aliasing and Anti-Aliasing Filters – Signal Bandwidth and System Bandwidth – Distortionless Transmission through Systems – Ideal Low Pass, High Pass and Band Pass Filters.

AI Integration: AI-based spectral analysis of signals, Intelligent detection of aliasing and sampling effects, AI-assisted analysis of CTFT and DTFT characteristics, Intelligent sampling and signal reconstruction

Free/Open-Source AI Tools:

- Python (Librosa, SciPy, NumPy)
- Scikit-learn, Matplotlib, NumPy
- TensorFlow, PyTorch
- Audacity

UNIT III: Laplace Transform, Z-Transform and System Analysis

Laplace Transform: Definition and Region of Convergence (ROC) – Properties and Inverse Laplace Transform – Transfer Function and System Response – Poles and Zeros in S-Plane – Stability Analysis and BIBO Stability – Solution of Differential Equations using Laplace Transform. Z-Transform: Definition and ROC – Properties of Z-Transform – Inverse Z-Transform Techniques – Solution of Difference Equations – Pole-Zero Analysis in Z-Plane – Stability of Discrete-Time Systems – Relationship among Fourier, Laplace and Z-Transforms.

AI Integration: AI-assisted pole-zero analysis in S-plane and Z-plane, AI-assisted analysis of transfer functions and system behavior, Intelligent stability prediction of continuous and discrete-time systems, Data-driven comparison of Fourier, Laplace, and Z-Transforms

Free/Open-Source AI Tools:

- Python (Control Systems Library)
- Scikit-Learn
- GNU Octave Control Package
- MATLAB Alternative: Scilab Xcos
- Keras

UNIT IV: Discrete Fourier Transform and Digital Filter Structures

Discrete Fourier Series (DFS) – Discrete Fourier Transform (DFT) – Properties of DFT – Circular and Linear Convolution using DFT – Fast Fourier Transform (FFT) – Radix-2 Decimation-in-Time FFT – Radix-2 Decimation-in-Frequency FFT – Inverse FFT and FFT for Composite N – Realization Structures of Digital Filters – FIR and IIR Filter Structures – Direct Form-I and II Structures – Cascade and Parallel Realization – Transposed Structures – Frequency Domain Representation of Discrete-Time Systems.

AI Integration: AI-assisted FFT spectrum analysis, Machine learning-based digital filter realization analysis, Intelligent FFT-based signal classification, AI-assisted convolution and frequency-domain processing

Free/Open-Source AI Tools:

- Python (TensorFlow, PyTorch)
- GNU Octave Signal Package
- Scikit-learn
- Jupyter Notebook

UNIT V: Digital Filter Design and Multirate DSP

Analog Filter Approximations: Butterworth and Chebyshev Filters – Design of IIR Digital Filters – Bilinear Transformation and Impulse Invariance Techniques – Characteristics of FIR Filters – FIR Filter Design using Window Techniques – Frequency Sampling Method – Comparison of FIR and IIR Filters – Multirate Digital Signal Processing – Decimation and Interpolation – Polyphase Structures – Nyquist Filters – Spectral Analysis of Non-Stationary Signals – Applications of DSP in Audio, Communication and Biomedical Engineering.

AI Integration: AI-assisted design optimization of FIR and IIR digital filters, Machine learning-based spectral analysis of non-stationary signals, AI-assisted communication signal analysis and enhancement, Data-driven comparison of Butterworth and Chebyshev filters

Free/Open-Source AI Tools:

- Python (MNE, SciPy, TensorFlow)
- GNU Radio
- Scikit-learn

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the fundamental concepts of signals, systems, Fourier series, and properties of continuous-time and discrete-time systems with AI-assisted visualization tools.	L1 (Remember) L2 (Understand)
CO2	Understand Fourier transform techniques, sampling theory, signal reconstruction, and filtering concepts using conventional and AI-based analytical methods.	L2 (Understand)
CO3	Apply Laplace Transform, Z-Transform, DFT, FFT, and digital filter realization techniques for signal and system analysis using computational and AI-assisted tools.	L3 (Apply)
CO4	Analyze stability, spectral characteristics, convolution operations, and filter performance of continuous-time and discrete-time systems using classical and AI-enabled approaches.	L4 (Analyze)
CO5	Design FIR/IIR digital filters, multirate DSP systems, and intelligent signal processing models for engineering applications using conventional and AI-based techniques.	L5 (Evaluate) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**L1, L2**	3	2	1	–	–	–	–	–	–	1	–	3	1	1
CO2	**L2**	3	3	2	2	–	–	–	–	1	1	–	3	2	2
CO3	**L3**	3	3	3	3	1	–	–	–	1	1	–	3	3	2
CO4	**L4**	3	3	2	3	2	–	1	–	1	1	–	3	2	3
CO5	**L5, L6**	3	3	3	3	3	2	1	–	2	2	1	3	3	3

Textbooks

1. Signals and Systems, A.V. Oppenheim, A.S. Willsky and S.H. Nawab, PHI, 2nd Edition, 2009.

2. Digital Signal Processing – A Computer Based Approach, Sanjit K. Mitra, Tata McGraw Hill, 3rd edition, 2009.

Reference Books

1. Principles of Linear Systems and Signals, B.P. Lathi, Oxford University Press, 2nd Edition, 2015.

2. Signals and Systems: A Primer with MATLAB, Matthew N.O. Sadiku and Warsame H. Ali, CRC Press, 2016.

3. Signals & Systems, Simon Haykin and Van Veen, Wiley, 2nd Edition, 2005.

4. Digital Signal Processing: Principles, Algorithms and Applications, John G. Proakis, Dimitris G. Manolakis, Pearson Education/PHI, 2007.

5. Schaum's Outline of Signals and Systems, Hwei Hsu, Fourth Edition, TMH, 2019.

6. Digital Signal Processing, Andreas Antoniou, TATA McGraw Hill, 2006.

7. A Text book on Digital Signal Processing, R.S. Kaler, M. Kulkarni, Umesh Gupta, I K International Publishing House Pvt. Ltd., 2009.

Web Resources & NPTEL Links

- NPTEL – Signals and Systems: https://nptel.ac.in/courses/108/106/108106151/
- NPTEL Lectures on Signals and Systems by Prof. K.S. Venkatesh, IIT Kanpur
- NPTEL Lectures on Signals and Systems by Prof. Aditya K. Jagannatham, IIT Kanpur

II Year II Sem

Course Code	AI ASSISTED CONTROL SYSTEMS LAB	L	T	Pr	P	Credits
26AEE02401P	II Year II Sem	0	0	0	3	1.5
Prerequisites:	Control Systems, AI Tools & Applications					

Course Objectives

- Perform experiments on control systems and analyze system behaviour.
- Evaluate system performance using time and frequency domain techniques.
- Design and implement controllers for real-time systems.
- Apply modern tools and AI techniques in control experiments.
- Develop skills in experimentation, data analysis, and reporting.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the characteristics, transfer functions, stability concepts, and operational principles of control system components with AI-assisted visualization tools.	L1 (Remember) L2 (Understand)
CO2	Understand the effects of feedback, controller actions, compensation techniques, and PLC operations in control systems using conventional and AI-based methods.	L2 (Understand)
CO3	Apply time response analysis, stability analysis, PID controller design, and PLC programming techniques for control system applications using computational and AI-assisted tools.	L3 (Apply)
CO4	Analyze system stability, controllability, observability, servo motor performance, and controller characteristics using classical and AI-enabled analytical approaches.	L4 (Analyze)
CO5	Design intelligent control solutions, PLC-based automation systems, and AI-assisted monitoring and control models for industrial applications.	L5 (Evaluate) L6 (Create)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3

CO1	**L1, L2**	3	2	–	–	2	–	–	–	–	1	–	3	1	1
CO2	**L2**	3	3	1	1	3	–	–	–	1	1	–	3	2	1
CO3	**L3**	3	3	2	2	3	–	–	–	1	1	–	3	3	2
CO4	**L4**	3	3	3	3	3	–	1	–	1	1	–	3	3	3
CO5	**L5, L6**	2	2	3	2	3	1	1	–	2	2	1	2	2	3

List of Experiments

Any 10 experiments of the following are required to be conducted:

- Time response of Second order system
- Characteristics of Synchros
- Effect of feedback on DC servo motor
- Transfer function of DC Machine
- Effect of P, PD, PI, PID Controller on a second order system
- Lag and lead compensation – Magnitude and phase plot
- Design of PID Controller for Temperature control using process simulator
- Characteristics of magnetic amplifiers
- Characteristics of AC servo motor
- Verification of system stability using Root locus, Bode Plot & Nyquist Plot
- Verification of time response analysis for a given system
- Verification of controllability and observability for a given system
- Programmable logic controller – Study and verification of truth tables of logic gates, simple Boolean expressions
- PLC Based Temperature Control
- Speed Control of DC Motor Using PLC

AI Integration:

Minimum two of the following AI experiments are to be conducted:

- ML-based prediction of time response parameters
- Smart PLC using AI decision-making
- Neural network modelling of servo motor
- AI-based PID tuning
- Optimization based compensator design
- Intelligent temperature control (Fuzzy / ANN)
- Time response prediction using deep learning

Recommended Free & Open-Source AI/Computational Tools

Control & Simulation:

- GNU Octave
- Scilab + Xcos
- OpenModelica

AI / ML Tools:

- Python (NumPy, SciPy, scikit-learn)
- TensorFlow / PyTorch
- Jupyter Notebook / Google Colab

Optimization / Advanced AI:

- DEAP (Genetic Algorithm)
- PyGAD
- OpenAI Gym (Reinforcement Learning)

II Year II Sem

Course Code	AI ASSISTED CONTROL OF INDUCTION AND SYNCHRONOUS MACHINES LAB	L	T	Pr	P	Credits
26AEE02403P	II Year II Sem	0	0	0	3	1.5
Prerequisites:	AI-Assisted Control of Induction and Synchronous Machines					

Course Objectives

- Perform experimental investigations on induction motors, synchronous machines and AC series motors.
- Analyze experimental results to determine performance characteristics such as efficiency, regulation and power factor.
- Study different testing methods for evaluating electrical machine performance.
- Utilize computational tools for visualization and analysis of experimental data.
- Apply AI-based tools for intelligent monitoring, prediction and fault detection in electrical machines.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Remember and explain the operating principles, characteristics, testing procedures, and performance parameters of induction motors, synchronous motors, alternators, and AC series motors used in laboratory experiments.	L1 (Remember) L2 (Understand)
CO2	Apply experimental methods to conduct brake tests, circle diagram analysis, speed control, power factor improvement, voltage regulation tests, efficiency determination, and parallel operation of AC electrical machines.	L3 (Apply)
CO3	Analyze the performance characteristics, efficiency, voltage regulation, speed–torque behaviour, synchronous reactance, and operating conditions of AC machines using measured and simulated experimental data.	L4 (Analyze)
CO4	Apply and analyze AI and machine learning techniques such as regression models, neural networks, anomaly detection, and predictive analytics for efficiency estimation, fault diagnosis, voltage regulation prediction, and machine condition monitoring.	L3 (Apply) L4 (Analyze)
CO5	Design AI-assisted monitoring, predictive maintenance, and intelligent performance evaluation models for rotating electrical machines using experimental data visualization, pattern	L5 (Evaluate) L6 (Create)

	recognition, and open-source computational tools.	

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**L1, L2**	3	2	1	3	2	1	–	2	1	–	2	3	1	1
CO2	**L3**	3	3	2	3	2	1	–	2	1	–	2	3	2	1
CO3	**L4**	3	3	2	3	2	1	–	2	1	–	2	3	2	2
CO4	**L3, L4**	2	2	1	2	3	1	–	2	2	–	3	2	2	2
CO5	**L5, L6**	2	2	2	2	3	2	–	2	2	1	3	2	3	3

List of Experiments

Any 10 of the following experiments are to be conducted:

- Brake test on three phase Induction Motor
- Circle diagram of three phase induction motor
- Speed control of three phase induction motor by V/f method
- Equivalent circuit of single-phase induction motor
- Power factor improvement of single-phase induction motor by using capacitors
- Load test on single phase induction motor
- Regulation of a three-phase alternator by synchronous impedance & MMF methods
- Regulation of three-phase alternator by Potier triangle method
- V and Inverted V curves of a three-phase synchronous motor
- Determination of Xd, Xq & Regulation of a salient pole synchronous generator
- Determination of efficiency of three phase alternator by loading with three phase induction motor
- Parallel operation of three-phase alternator under no-load and load conditions
- Determination of efficiency of a single-phase AC series Motor by conducting Brake test

AI Integration:

Minimum two of the following AI experiments are to be conducted:

- Machine learning models to predict efficiency of induction motors and alternators
- AI-based speed–torque characteristic prediction for induction motors
- Predictive maintenance models for electrical machines using vibration/current data
- Regression models to estimate voltage regulation of alternators
- AI-based analysis of V and inverted V curves of synchronous motors
- Neural network estimation of synchronous reactance (Xd and Xq)
- Data visualization and pattern recognition for experimental machine characteristics

Web Resources & NPTEL Links

- Virtual Labs – Electrical Machines: https://em-coep.vlabs.ac.in/List%20of%20experiments.html
- NPTEL – Electrical Machines: https://nptel.ac.in/courses/108105017
- MIT OpenCourseWare – Electric Machines: https://ocw.mit.edu/courses/electric-machines/
- Python for Engineering Applications: https://scipy.org
- Machine Learning with Python: https://scikit-learn.org/stable/tutorial/index.html
- OpenModelica Simulation Course: https://openmodelica.org/education

II Year II Sem

Course Code	ELECTRICAL SIMULATION TOOLS	L	T	Pr	P	Credits
26AEE02406S	II Year II Sem	1	0	0	2	2
Prerequisites:	Electrical Circuits, Electrical Machines, AI Tools & Applications					

Course Objectives

- Introduce the fundamentals of Electronic Design Automation (EDA), P-SPICE simulation techniques, and electrical circuit analysis methods.
- Develop the ability to analyze DC, AC, and power electronic circuits using P-SPICE simulation tools.
- Provide knowledge of MATLAB programming, numerical computation, visualization, and engineering problem solving.
- Familiarize students with MATLAB/Simulink applications in control systems, signal processing, neural networks, and fuzzy logic systems.
- Develop practical skills in E-CAD tools for schematic capture, PCB layout design, routing, and integration of simulation with PCB development.

Course Content

UNIT I: P-SPICE Fundamentals and DC Circuit Simulation

Introduction to Electronic Design Automation (EDA) – Overview of P-SPICE environment – Circuit components, libraries, and schematic capture – DC analysis: Ohm's Law and Kirchhoff's Laws, Nodal and Mesh Analysis, Voltage and Current Divider Circuits, DC Sweep and Bias Point Analysis – Network theorems using P-SPICE: Superposition Theorem, Thevenin's Theorem & Norton's Theorem.

UNIT II: P-SPICE for AC and Power Electronics Applications

AC circuit simulation using P-SPICE – Transient and Frequency Response Analysis – RLC circuits and resonance – Introduction to semiconductor devices: Diodes, BJT, MOSFET – Harmonic analysis and waveform visualization.

UNIT III: MATLAB Fundamentals for Electrical Engineering Applications

Introduction to MATLAB environment – MATLAB commands, scripts, and functions – Matrix operations and array manipulation – Plotting and visualization tools – Numerical methods in MATLAB – Solving electrical engineering equations using MATLAB – Applications in circuit analysis and signal representation.

UNIT IV: MATLAB for Control Systems and Signal Processing

Introduction to Simulink – Modeling of dynamic systems – Transfer functions and state-space models – Time-domain analysis: Step response, Impulse response – Frequency-domain analysis: Bode plots, Nyquist plots – MATLAB applications in electrical drives and power systems – Neural Network Toolbox: Structure of neurons, transfer functions, representation of networks, simulating networks, Adaline, Backpropagation, self-organizing networks – Fuzzy logic toolbox: demonstration of fuzzy logic, Membership functions, Rule editor, rule viewer and surface viewer.

UNIT V: E-CAD Tools for PCB Design and Electrical Layouts

Introduction to E-CAD tools – PCB design flow and methodology – Schematic capture and netlist generation – PCB layout fundamentals: Layer management, Routing techniques, Ground planes – Design Rule Check (DRC) – Industrial PCB Design Standards – Integration of Simulation and PCB Design.

Course Outcomes (COs)

CO	Course Outcome Statement	Bloom's Level
CO1	Explain the fundamentals of P-SPICE environment, DC circuit analysis, and network theorems for electrical circuit simulation.	L1 (Remember)
CO2	Interpret AC circuit behavior, semiconductor device characteristics, and transient/frequency responses using P-SPICE tools.	L2 (Understand)
CO3	Apply MATLAB programming concepts, matrix operations, and numerical methods for solving electrical engineering problems.	L3 (Apply)
CO4	Analyze control systems, signal processing models, neural networks, and fuzzy logic applications using MATLAB/Simulink.	L4 (Analyze)
CO5	Design PCB layouts and electrical schematics using E-CAD tools by following industrial design standards and verification techniques.	L4 (Analyze) L5 (Evaluate)

CO – PO Articulation Matrix

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

CO	Bloom's Level	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	**L1**	3	2	–	1	3	–	–	–	1	–	2	3	1	1
CO2	**L2**	3	3	1	2	3	–	–	1	1	–	2	3	2	1
CO3	**L3**	3	2	2	2	3	–	–	1	1	–	2	3	2	2
CO4	**L4**	3	3	3	3	3	1	1	2	2	2	3	3	3	3
CO5	**L4, L5**	3	3	3	2	3	2	1	2	2	2	3	3	3	3

List of Experiments

- Nodal and Mesh analysis of a given electrical network

- Calculation of Maximum Power dissipation of a given circuit
- Obtain transient response of the given circuit
- Determination of transfer function, Poles and zeros of a given circuit
- Determination of frequency response of a given circuit
- Determination of transmission parameters of a given cascaded network
- Generation of Full wave rectifier waveform
- Determination of diode characteristics
- Determination of output characteristics of BJT and NPN transistor
- Design of a single phase transformer with the given specifications
- Design of an Induction motor with the given specifications
- Determination of mmf required for the air gap of a machine
- Neural Network-Based Prediction System
- Fuzzy Logic-Based Intelligent Control System
- Hybrid Neuro-Fuzzy System
- Design of PCB layout for regulated power supply
- Schematic capture and PCB routing using E-CAD tool

Textbooks

1. Simulation of Electric and Electronic Circuits using PSPICE by M.H. Rashid, PHI Publications.
2. MATLAB: The Language of Technical Computing, MathWorks Inc., 2005.
3. PCB Design Tutorial by David L. Jones.

II Year II Sem

Course Code	TECHNICAL REPORT WRITING & INTELLECTUAL PROPERTY RIGHTS	L	T	Pr	P	Credits
26AHS0302M	Common to All Branches	2	0	0	0	0

Course Overview

This audit course equips engineering students with professional technical writing competencies and a comprehensive understanding of Intellectual Property Rights. The course follows a building-block approach: foundational concepts of technical communication are introduced first, followed by progressive application in report writing, research documentation, and IP strategy. AI-powered writing tools are embedded throughout to enhance productivity while reinforcing the principles of academic integrity.

Detailed Syllabus

UNIT 1: Foundations of Technical Communication

Core Concepts:

- Technical Communication vs. General Communication: Precision, objectivity, and audience-awareness
- Types of Technical Documents: Reports, Proposals, Manuals, Specifications, Abstracts
- Writing Process for Technical Documents: Planning, Drafting, Revising, Proofreading
- Audience Analysis: Primary and secondary audiences; adapting tone and technicality
- Language in Technical Writing: Active vs. Passive voice, Nominalisations, Clarity and conciseness

AI Integration:

- AI Tool: Grammarly (free) – analyse a technical paragraph for passive voice overuse and nominalisation
- AI Tool: Hemingway Editor – paste a technical section and improve readability score to Grade 10 or below
- Activity: Compare two versions of a technical document (before and after AI editing) and document changes

UNIT 2: Technical Report Writing

Core Concepts:

- Structure of a Formal Report: Title page, Abstract, Table of Contents, Introduction, Methodology, Results, Discussion, Conclusion, References, Appendices
- Abstract Writing: Informative vs. Descriptive abstracts; 150–250 word limit
- Literature Review: Searching, Evaluating, and Synthesising sources
- Data Presentation: Tables, Figures, Charts – placement, numbering, and captioning
- Citation and Referencing: IEEE, APA, and MLA styles; avoiding plagiarism

AI Integration:

- AI Tool: Elicit.org (free) – AI-powered literature search; find and summarise 5 relevant papers

- AI Tool: Zotero (open-source reference manager) – manage citations and auto-generate bibliographies
- AI Tool: ChatGPT – generate a first-draft abstract; critique and rewrite to meet technical writing standards
- Plagiarism Check: Use Turnitin (institutional access) or Copyleaks (free tier) to verify originality

UNIT 3: Research Proposals and Scientific Writing

Core Concepts:

- Research Proposal Structure: Background, Problem Statement, Objectives, Methodology, Timeline, Budget
- Scientific Paper Format: IMRAD Structure (Introduction, Methods, Results and Discussion)
- Writing Results vs. Discussion: Factual reporting vs. Interpretation
- Peer Review Process: Understanding blind review, revision, and resubmission
- Conference Papers vs. Journal Articles: Scope, length, and submission guidelines

AI Integration:

- AI Tool: Semantic Scholar (free) – search for open-access papers in your domain; analyse IMRAD structure
- AI Tool: Writefull (free academic writing app) – get AI feedback on scientific language in a draft methods section
- Activity: Write a 1-page research proposal for a small project; use AI for structural review and language polishing

UNIT 4: Intellectual Property Rights – Fundamentals

Core Concepts:

- Introduction to IPR: Definition, importance, and types – Patents, Trademarks, Copyrights, Trade Secrets, Geographical Indications
- Patent System in India: Indian Patent Act 1970, Patent Office, Patent application process
- Patentability Criteria: Novelty, Inventive Step (Non-obviousness), Industrial Applicability
- Patent Search: Prior art search using databases
- Copyright in Engineering: Software copyright, Open-source licenses (GPL, MIT, Apache)

AI Integration:

- AI Tool: Google Patents (free) – conduct a prior art search on a given invention idea
- AI Tool: Lens.org (free, open-source patent database) – search and analyse patent documents
- Activity: Identify 3 patents related to a chosen engineering domain and analyse their claims section
- Discussion: How does AI help in patent drafting? Use AI tools like PatSnap or Examiner (where accessible)

UNIT 5: Patent Drafting and IP Strategy

Core Concepts:

- Structure of a Patent Application: Title, Abstract, Background, Summary, Detailed Description, Claims, Drawings
- Types of Claims: Independent and Dependent claims; method, product, system claims
- Filing a Provisional Patent Application (PPA) in India
- IP Strategy for Startups and Engineers: Trade secrets vs. patents, licensing, technology transfer
- Case Studies: Famous patent disputes (Apple vs. Samsung, Monsanto seed patents, pharmaceutical patents)

AI Integration:

- AI Tool: ChatGPT – draft a set of patent claims for a simple mechanical invention; critique for clarity and scope
- AI Tool: IP India Online Portal (ipindia.gov.in) – navigate and understand the e-filing system
- Capstone Project: Write a complete patent disclosure document for a project-based invention idea
- Activity: Use AI to convert a project report executive summary into a patent abstract; compare the two

Course Outcomes

CO	Course Outcome Statement	Bloom's Level	Bloom's Domain
CO1	Recall and understand the principles, types, and structure of technical reports, and explain the fundamentals of Intellectual Property Rights (IPR).	L1 – Remember / L2 – Understand	Cognitive
CO2	Apply the principles of technical writing to produce well-structured laboratory reports, project reports, and technical proposals.	L3 – Apply	Cognitive
CO3	Analyse technical documents for clarity, coherence, accuracy, and compliance with formatting standards using AI-powered review tools.	L4 – Analyse	Cognitive
CO4	Evaluate patent documents, prior art, and IP claims to assess novelty and inventiveness of an engineering innovation.	L5 – Evaluate	Cognitive
CO5	Create a complete technical report and draft a patent disclosure for an engineering project, integrating AI tools for writing assistance.	L6 – Create	Cognitive & Psychomotor

CO-PO Mapping Matrix

Scale: 3 = High, 2 = Medium, 1 = Low, '-' = No correlation

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	1	1	1	1	1	2	1	3	2	2	2	2
CO2	2	2	2	2	2	1	1	2	2	3	2	3	2	2
CO3	2	3	2	2	3	1	1	2	2	3	2	3	2	2
CO4	2	3	2	3	2	2	2	3	2	2	2	3	3	3
CO5	2	2	3	2	3	1	1	2	3	3	3	3	3	3
Average	1.8	2.2	2.0	2.0	2.2	1.2	1.2	2.2	2.0	2.8	2.2	2.8	2.4	2.4

Text Books

1. T1. Markel, Mike & Selber, Stuart. (2022). Technical Communication (13th ed.). Bedford/St. Martin's, New York.
2. T2. Ahuja, B.N. & Ahuja, S.S. (2021). Intellectual Property Rights: Patents, Trade Marks, Copyrights and Related Topics (4th ed.). Khanna Publishers, New Delhi.

Reference Books

1. R1. Lannon, John & Gurak, Laura. (2020). Technical Communication (15th ed.). Pearson Education.
2. R2. Anderson, Paul V. (2019). Technical Communication: A Reader-Centred Approach (9th ed.). Cengage.
3. R3. Meganathan, R. (2019). Intellectual Property Rights for Engineers. Universities Press (India).
4. R4. Cornish, William R. & Llewelyn, David. (2019). Intellectual Property: Patents, Copyright, Trademarks and Allied Rights (9th ed.). Sweet & Maxwell.
5. R5. Eisenberg, Anne. (2018). Effective Technical Communication. Tata McGraw-Hill.
6. R6. World Intellectual Property Organization (WIPO). (2022). WIPO Intellectual Property Handbook. WIPO Geneva.

Web Resources

- IP India – Patent Office: www.ipindia.gov.in
- Google Patents: patents.google.com
- Lens.org – Free Patent and Scholar Search: www.lens.org
- WIPO – Patent Resources: www.wipo.int
- Purdue OWL – Technical Writing: owl.purdue.edu/owl/subject_specific_writing/professional_technical_writing

YouTube / Video Resources

- YouTube: NPTEL – Technical English for Engineers – www.youtube.com/c/nptel
- YouTube: IP India – Patent Filing Guide – search on YouTube
- YouTube: WIPO – Intellectual Property for Engineers – www.youtube.com/@WIPOChannel
- YouTube: How to Write a Patent Application – PatentFile – search on YouTube

YouTube: Scholarly Writing Tips – Academic English – search on