# SRI SAI
## INSTITUTE OF TECHNOLOGY AND SCIENCE
## AUTONOMOUS

ESTD.:2001

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

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

# B.Tech.
## Mechanical 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)
Estd. 2001
Masapeta, RAYACHOTY - 516 270, Annamayya District, A.P.

# B. Tech (Regular-Full time)

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

# MECHANICAL 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)

## B.Tech. – I Year I Semester

| S.No. | Course code | Title | L/D | T | Pr | P | Credits |
|---|---|---|---|---|---|---|---|
| 1 | 26AHS0104T | Applied Chemistry for Mechanical Engineers | 3 | 0 | 0 | 0 | 3 |
| 2 | 26AHS0102T | Linear Algebra & Calculus | 3 | 0 | 2 | 0 | 4 |
| 3 | 26ACS5101T | Design Thinking | 1 | 0 | 4 | 0 | 3 |
| 4 | 26ACS5102T | Computational Thinking and Problem Solving with Python | 3 | 0 | 0 | 0 | 3 |
| 5 | 26ACS5103T | AI Tools and Applications | 1 | 0 | 4 | 0 | 3 |
| 6 | 26AME3101P | Engineering Workshop | 0 | 0 | 0 | 3 | 1.5 |
| 7 | 26AHS0104P | Applied Chemistry for Mechanical Engineers Lab | 0 | 0 | 0 | 3 | 1.5 |
| 8 | 26ACS5102P | Computational Thinking and Problem Solving with Python Lab | 0 | 0 | 0 | 3 | 1.5 |
| 9 | 26AHS0101L | NSS/NCC/Scouts and Guides / Community Service | 0 | 0 | 0 | 1 | 0.5 |
| | | **Total** | **11** | **0** | **10** | **10** | **21** |

## B.Tech. – I Year II Semester

| S.No. | Course code | Title | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|---|
| 1 | 26AHS0201T | Communicative English | 2 | 0 | 0 | 0 | 2 |
| 2 | 26AHS0201T | Physics for Mechanical Systems | 3 | 0 | 0 | 0 | 3 |
| 3 | 26AHS0203T | Differential Equations & Vector Calculus | 3 | 0 | 2 | 0 | 4 |
| 4 | 26AME3202T | **Professional core - I**<br>• Engineering Mechanics | 3 | 0 | 0 | 0 | 3 |
| 5 | 26AME3201T | Engineering Graphics | 1 | 0 | 4 | 0 | 3 |
| 6 | 26AME3203P | **Department Specific Workshop**<br>• Basic Simulation Lab | 0 | 0 | 0 | 2 | 1 |
| 7 | 26AHS0201P | Physics for Mechanical Systems Lab | 0 | 0 | 0 | 3 | 1.5 |
| 8 | 26AHS0201P | Communicative English Lab | 0 | 0 | 0 | 2 | 1 |
| 9 | 26AHS0201L | Health and wellness, Yoga and Sports | 0 | 0 | 0 | 1 | 0.5 |
| 10 | 26ACS5201M | Cybersecurity Awareness for Engineers | 2 | 0 | 0 | 0 | 0 |
| | | **Total** | **14** | **0** | **6** | **8** | **19** |

**B.Tech.– II Year I Semester**

| S.No. | Course code | Title | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|---|
| 1 | 26AHS0303T | Complex Variables and Probability Theory | 3 | 0 | 0 | 0 | 3 |
| 2 | 26AHS0302T | Universal Human Values – Understanding Harmony and Ethical Human Conduct | 3 | 0 | 0 | 0 | 3 |
| 3 | 26AMB2301T | Managerial Economics and Financial Analysis | 2 | 0 | 0 | 0 | 2 |
| 4 | 26AEC304T | **Engineering Course**<br>• Basic Electricals and Electronics Engineering | 3 | 0 | 0 | 0 | 3 |
| 5 | 26AME3301T | **Professional core - II**<br>• Material Science and Metallurgy | 2 | 0 | 2 | 0 | 3 |
| 6 | 26AME3302T | **Professional core - III**<br>• Mechanics of Solids | 2 | 0 | 2 | 0 | 3 |
| 7 | 26AEC304P | **Engineering Course**<br>• Basic Electricals and Electronics Engineering Lab | 0 | 0 | 0 | 2 | 1 |
| 8 | 26AME3303P | **Professional core - II Lab**<br>• Engineering Mechanics Lab | 0 | 0 | 0 | 3 | 1.5 |
| 9 | 26AME3302P | **Professional core - III Lab**<br>• Mechanics of Solids And Material Science Lab | 0 | 0 | 0 | 3 | 1.5 |
| 10 | 26AME3304S | **Skill Enhancement course-I**<br>• Computer Aided Machine Drawing | 1 | 0 | 0 | 2 | 2 |
| 11 | 26AHS0301M | Environmental Science | 2 | 0 | 0 | 0 | 0 |
| | | **Total** | **18** | **0** | **4** | **10** | **23** |

**B.Tech. 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 | 26AME3402T | **Professional core - IV**<br>• Fluid Mechanics and Hydraulic Machinery | 2 | 0 | 2 | 0 | 3 |
| 4 | 26AME3403T | **Professional core - V**<br>• Manufacturing Processes | 2 | 0 | 2 | 0 | 3 |
| 5 | 26AME3404T | **Professional core - VI**<br>• Kinematics of Machinery | 2 | 0 | 2 | 0 | 3 |
| 6 | 26AME3405T | **Professional core - VII**<br>• Thermodynamics | 2 | 0 | 2 | 0 | 3 |
| 7 | 26AME3402P | **Professional core – IV lab**<br>• Fluid Mechanics and Hydraulic Machinery Lab | 0 | 0 | 0 | 3 | 1.5 |
| 8 | 26AME3402P | **Professional core - V Lab**<br>• Manufacturing Processes Lab | 0 | 0 | 0 | 3 | 1.5 |
| 9 | 26AEC404S | **Skill Enhancement course-II**<br>• Embedded Systems and IoT Lab | 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 | | | | | | |

# APPLIED CHEMISTRY FOR MECHANICAL ENGINEERING

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0104T | I Year II Sem | 3 | 0 | 0 | 0 | 3 |

**Branch:** Mechanical Engineering (ME) | Regulation: JNTUA R26 (per source file name; not stated as a labelled field in the source document.)

## Course Objectives

- Understand corrosion mechanisms and surface engineering in mechanical systems.
- Analyze fuels, combustion, and energy conversion processes.
- Study lubrication chemistry and material behavior under friction.
- Explore engineering materials including metals, polymers, and composites.
- Apply AI/ML tools for prediction, optimization, and sustainability in mechanical engineering.

## Course Content

### UNIT 1: Corrosion chemistry and protective methods (Duration: 3 weeks/9Hrs)

**UNIT Objectives:**

- To understand corrosion mechanisms and prevention in mechanical components
- To comprehend electroplating and surface treatment processes for enhancing component properties

**UNIT Outcomes:**

- Students can analyze corrosion types in mechanical systems and recommend prevention strategies
- Students can design surface treatment processes for improving wear resistance and corrosion protection

**Content:**

- Chemical (Atmospheric corrosion: rusting of iron and steel) and Electrochemical theories of corrosion: anodic and cathodic reactions, electrochemical series.
- Types of corrosion: uniform, galvanic, pitting, crevice, intergranular, stress corrosion.
- Corrosion prevention: cathodic protection, protective coatings-Galvanizing: hot-dip galvanizing, electro galvanizing, Sherardizing.
- Electroplating: principles, process parameters (current density, temperature, pH) Ex: Copper plating, Nickel plating
- Anodizing: aluminium anodizing, hard anodizing, coloring
- Surface hardening: carburizing, nitriding, carbonitriding, flame hardening
- Conversion coatings: phosphating and chromating.

**AI Integration:** Machine learning models for corrosion prediction based on environmental and material data. Image-based corrosion detection using computer vision.

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

### UNIT 2: Energy Chemistry (Fuels and Combustion) (Duration: 3 weeks/9Hrs)

**UNIT Objectives:**

- To understand the chemistry of fuels and combustion processes for mechanical systems
- To evaluate energy conversion and alternative fuels for automotive applications

**UNIT Outcomes:**

- Students can analyze fuel properties and combustion efficiency for engines and boilers
- Students can assess alternative fuels and their suitability for mechanical engineering applications

**Content:**

- Classification of fuels: solid, liquid, gaseous fuels, calorific value: HCV and LCV, Dulong's formula, problems. Calorific value determination: bomb calorimeter, problems.
- Solid fuels (Coal): classification (anthracite, bituminous, lignite, peat), analysis (proximate, ultimate), carbonization

- Liquid fuels (Petroleum oil): composition, refining processes (fractional distillation and cracking- thermal and catalytic), Synthetic fuels: Fischer-Tropsch synthesis
- Green fuels- Green hydrogen, Hythane.
- Knocking in IC engines (petrol engines): octane number, anti-knock additives: Tetra ethyl Lead (TEL), Methyl Tertiary Butyl Ether (MTBE), ethanol.
- Knocking in diesel engines: cetane number, ignition quality, anti-knock additives.
- Combustion -- problems.

**AI Integration:** AI models to optimize fuel efficiency and combustion performance. Data analysis of engine parameters using regression models.

**Free/Open-Source AI Tools:** Python (NumPy, Pandas), TensorFlow, Google Colab

## UNIT 3: Chemistry of Lubricants (Duration: 3 weeks/9Hrs)

**UNIT Objectives:**

- To understand the science of friction, wear, and lubrication in mechanical systems
- To evaluate lubricant properties and selection criteria for various applications

**UNIT Outcomes:**

- Students can explain friction and wear mechanisms and their impact on machine components
- Students can select appropriate lubricants based on chemical composition and operating conditions

**Content:**

- Friction and wear, seizure (soiling and rolling friction) functions of lubricants.
- Mechanisms of lubrication a. Fluid-film or thick-film or hydrodynamic lubrication, b. boundary film lubrication or thin-film lubrication, c. Extreme-pressure lubrication
- Lubricants: classification
- Solid lubricants: graphite, molybdenum disulfide (MoS2), Poly tetra fluoro ethylene (PTFE).
- Semi-solid lubricants: Greases-composition (base oil + thickener + additives), National Lubricating Grease Institute (NLGI) grades, applications
- Liquid lubricants: animal and vegetable oils, mineral or petroleum oils, blended oils.
- Properties of lubricants (Definition and significance)

**AI Integration:** Predictive maintenance using AI for lubricant degradation analysis. Data-driven selection of lubricants based on operating conditions.

**Free/Open-Source AI Tools:** Python, Excel AI tools, scikit-learn.

## UNIT 4: Engineering Materials and Composites (Duration: 3 weeks/9Hrs)

**UNIT Objectives:**

- To understand the chemistry and properties of materials used in mechanical engineering
- To evaluate composite materials and their processing for mechanical applications

**UNIT Outcomes:**

- Students can explain structure-property relationships in metals, polymers, and ceramics
- Students can design and fabricate composite materials for specific mechanical applications

**Content:**

- Ferrous alloys: composition and applications of carbon steels, alloy steels, stainless steels, cast irons.
- Heat treatment: annealing, normalizing, hardening, tempering, age hardening
- Polymer chemistry: types of polymerizations, examples, thermoplastics, thermosets,
- Engineering polymers: Preparation and engineering applications of Nylon, polycarbonate, Acrylonitrile Butadiene styrene (ABS), Polyether ether ketone( PEEK), Polytetrafluoroethylene (PTFE)
- Polymer processing: Injection moulding, Extrusion moulding, Compression moulding
- Elastomers- Natural rubber, vulcanisation, synthetic rubbers (BUNA-S and BUNA-N)
- Ceramics: alumina, silicon carbide, silicon nitride, zirconia and applications
- Refractories: characteristics, classification and properties of refractories.
- Composite materials: classification and applications.

  **AI Integration:** AI-based material property prediction and selection. Machine learning for composite optimization and performance analysis.

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

## UNIT 5: Chemistry of Green Engineering (Duration: 3 weeks/9Hrs)

**UNIT Objectives:**

- To understand principles of green chemistry for sustainable engineering practices
- To evaluate environmental impacts of construction materials and processes

**UNIT Outcomes:**

- Students can apply green chemistry principles to select sustainable materials and processes.
- Students can assess environmental chemistry aspects relevant to engineering projects.

**Content:**

- Principles of green chemistry and their applications
- Sustainable construction materials: geopolymer concrete, recycled aggregates, industrial by-products (fly ash, Ground Granulated Blast-furnace Slag (GGBS)
- Air pollution: types, sources, effects; air quality index
- Water pollution: sources, effects and remedial methods.
- Chemistry of greenhouse gases and climate change
- Soil chemistry: pH, organic matter, contamination, remediation
- Solid waste management: classification, treatment, recycling in construction

**AI Integration:** AI-based environmental impact assessment and life-cycle analysis. Air quality prediction models.

**Free/Open-Source AI Tools:** OpenLCA, Python, Google Colab

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Analyze corrosion and recommend protection methods | L4 |
| CO2 | Evaluate fuels and combustion processes | L3--L4 |
| CO3 | Apply lubrication principles in mechanical systems | L3 |
| CO4 | Analyze engineering materials and composites | L3--L4 |
| CO5 | Design sustainable solutions using green chemistry and AI tools | L5--L6 |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L4 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 1 | 1 | 2 | 1 | 2 | 3 | 2 |
| CO2 | L3-4 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 2 | 1 | 2 | 3 | 3 |
| CO3 | L3 | 2 | 2 | 3 | 2 | 3 | 2 | 2 | 1 | 2 | 2 | 1 | 2 | 3 | 3 |
| CO4 | L3-4 | 2 | 3 | 2 | 3 | 3 | 2 | 2 | 1 | 2 | 2 | 1 | 2 | 3 | 3 |
| CO5 | L5-6 | 3 | 2 | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 3 | 2 | 3 | 3 | 3 |

## Textbooks


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


## Reference Books


- Davis, M. L. (2020). Water and wastewater engineering: Design principles and practice (2nd ed.). McGraw-Hill Education.
- Shikha, A., & Sharma, R. K. (2019). Engineering chemistry. Krishna Prakashan Media.

- Dara, S. S., & Umare, S. S. (2018). Engineering chemistry (15th ed.). S. Chand Publishing.


## Online Learning Resources

- Society of Tribologists and Lubrication Engineers (STLE). (2023). Tribology and lubrication resources. https://www.stle.org/
- ASM International. (2023). Materials information and education. https://www.asminternational.org/
- Society of Automotive Engineers (SAE). (2023). Automotive engineering standards. https://www.sae.org/
- American Society of Mechanical Engineers (ASME). (2023). Mechanical engineering resources. https://www.asme.org/
- National Association of Corrosion Engineers (NACE). (2023). Corrosion prevention and control. https://www.nace.org/

## Recommended Free & Open-Source Chemistry / AI Tools

**Language & Environment:** Python, Google Colab – scripting and prototyping used across all five units.

**ML Libraries:** scikit-learn, TensorFlow, NumPy, Pandas, Orange Data Mining – model building, regression, and data analysis.

**Computer Vision:** OpenCV – image-based corrosion detection.

**Specialized Tools:** OpenLCA – environmental life-cycle analysis; Excel AI tools – lubricant data analysis and selection.

**I Year I Sem**

| Course Code | ENGINEERING MECHANICS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3202T | **Professional Core-1**<br>Common to Civil & Mechanical Branches | 3 | 0 | 0 | 0 | 3 |

**Prerequisites:** Engineering Mathematics (Linear Algebra & Calculus, Differential Equations)

## Course Objectives

- Introduce fundamental concepts of mechanics, force systems, and equilibrium with applications in structural and mechanical engineering.
- Develop ability to draw and interpret free body diagrams for coplanar, concurrent, and spatial force systems.
- Calculate centroid, centre of gravity, and moment of inertia for simple and composite bodies.
- Understand the fundamental concepts of stress, strain, elasticity, and the mechanical behaviour of materials under different loading conditions.
- Introduce AI/ML tools for simulation, analysis, and visualization of mechanical systems.

## Course Content

### UNIT I: Introduction, Force Systems & Friction

**Core Content:** Basic concepts of mechanics: scalar and vector quantities, Newton's laws, SI units. Coplanar concurrent forces: resultant by parallelogram law, triangle law, polygon law. Moment of a force: Varignon's theorem. Couples and resultants of force systems. Lami's theorem. Spatial force systems: vector representation, dot and cross products. Friction: limiting friction, Coulomb's laws of dry friction, coefficient of friction, angle of friction, cone of static friction, impending motion problems, wedge friction, screw jack.

**AI Integration:** Introduction to AI in Engineering: concept of data, machine learning (ML), and simulation. Using Python (NumPy) to solve force equilibrium problems computationally. Introduction to MATLAB/Scilab for vector force calculations. Visualizing force systems using Matplotlib/Python.

**Free/Open-Source AI Tools:** Python (NumPy, Matplotlib) – Google Colab (free); Scilab (free MATLAB alternative); GeoGebra (free online vector visualization)

### UNIT II: Equilibrium of Force Systems & Virtual Work

**Core Content:** Equilibrium conditions for coplanar systems: free body diagrams, equations of equilibrium ($\Sigma Fx=0$, $\Sigma Fy=0$, $\Sigma M=0$). Graphical methods: triangle and polygon of forces. Equilibrium of spatial systems. Analysis of plane trusses: method of joints, method of sections. Analysis of frames. Principle of virtual work: statement, applications to simple mechanisms and trusses.

**AI Integration:** Using Python/OpenSees (open source) for automated truss analysis. AI-based structural analysis: introduction to how FEM software uses matrix methods internally. Computer simulation of mechanism motion using Python. Visual debugging of FBDs using interactive tools.

**Free/Open-Source AI Tools:** OpenSees (free, open-source structural analysis); Python scipy.optimize for equilibrium solution; SAP2000 student edition (free); online truss solvers (free)

### UNIT III: Centroid, Centre of Gravity & Moment of Inertia

**Core Content:** Centroid of lines, areas, and volumes from first principles. Centroid of standard shapes: rectangle, triangle, circle, semicircle, quarter circle. Centroid of composite figures using tabular method. Centre of gravity of 3D bodies: prismatic bodies, cones, hemispheres, composite bodies. Pappus theorems and their applications. Area moment of inertia: polar moment, transfer theorem (parallel axis theorem). MOI of composite sections. Product of inertia. Mass moment of inertia for simple and composite bodies.

**AI Integration:** Python scripting for automated centroid and MOI calculations of composite cross-sections. Using spreadsheets (LibreOffice Calc) and Python Pandas for tabular calculations.

Introduction to how CAD software (AutoCAD/FreeCAD free) computes section properties. Application in AI-driven structural design optimization: cross-section selection.

**Free/Open-Source AI Tools:** Python (NumPy, Pandas) – Google Colab; FreeCAD (free, open-source CAD with section property computation); LibreOffice Calc (free)

### UNIT IV: Kinematics & Kinetics of Particles

**Core Content:** Kinematics: rectilinear motion – displacement, velocity, acceleration relations; curvilinear motion – Cartesian and polar coordinates; projectile motion. Kinetics: Newton's second law; D'Alembert's principle; work-energy method: kinetic energy, work done, power, efficiency; impulse-momentum method: linear and angular; impact: direct and oblique central impact, coefficient of restitution.

**AI Integration:** AI in Autonomous Vehicles: how kinematic equations underpin self-driving car algorithms. Python simulation of projectile motion with drag (real-world). ML-based trajectory prediction: introduction to LSTM for motion prediction. Robot path planning using kinematic constraints. Using Python (matplotlib.animation) for interactive motion visualization.

**Free/Open-Source AI Tools:** Python (NumPy, Matplotlib, SciPy) – Google Colab; Tracker (free video analysis software for kinematics experiments); Algodoo (free 2D physics simulation)

### UNIT V: Simple Stresses, Strains & Elastic Constants

**Core Content:** Elasticity and plasticity; types of stresses (tensile, compressive, shear) and strains (linear, lateral, shear, volumetric); Hooke's law; Poisson's ratio; elastic constants: E, G, K and their relationships; factor of safety; bars of varying cross-section; elongation under self-weight; bars in series and parallel; temperature stresses; stresses in composite bars. Introduction to principal stresses and strains: principal planes, Mohr's circle for 2D stress state.**StrainEnergy**Resilience–Gradual,sudden,impactandshockloadings–simpleapplications.

**AI Integration:** ML for material property prediction: using regression models to predict Young's modulus and Poisson's ratio from material composition data. Python-based stress-strain curve fitting using SciPy. Introduction to data-driven material models in structural analysis. Overview of physics-informed neural networks (PINNs) for material mechanics (conceptual).

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, Matplotlib) on Google Colab; Scilab (free); MIT OpenCourseWare stress analysis tools

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| **CO1** | Remember and explain the fundamental concepts of mechanics including Newton's laws, types of forces, equilibrium conditions, and Coulomb's friction laws, and demonstrate their relevance to Civil and Mechanical Engineering applications. | L1 (Remember)<br>L2 (Understand) |
| **CO2** | Apply the principles of force systems, free body diagrams, and equilibrium equations to solve coplanar, concurrent, and spatial force problems including trusses, frames, and friction-based mechanisms using analytical and computational methods. | L3 (Apply) |
| **CO3** | Analyze the geometric properties of engineering sections and bodies by calculating centroids, centre of gravity, and moments of inertia for composite shapes, and evaluate their significance in structural design using software tools. | L4 (Analyze) |
| **CO4** | Evaluate the motion and dynamic response of engineering systems using kinematics and kinetics of particles, applying work-energy and impulse-momentum principles and comparing analytical solutions with AI-based simulation outputs. | L4 (Analyze)<br>L5 (Evaluate) |
| **CO5** | Recall and explain the fundamental concepts of stress, strain, Hooke's law, elastic constants, and their inter-relationships, and identify the mechanical behaviour of engineering materials under axial, thermal, and composite loading conditions. | L1 (Remember)<br>L2 (Understand) |

## CO – PO Articulation Matrix

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

| CO | Bloom's Level | PO1 Eng. Knowledge | PO2 Problem Analysis | PO3 Design/ Dev. | PO4 Investigation | PO5 Modern Tools | PO6 Engineer & Society | PO7 Environ. & Sustain. | PO8 Ethics | PO9 Team Work | PO10 Commun. | PO11 PrjMgmt& Finance |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L1, L2 | 3 | 2 | 1 | - | - | - | - | - | - | 2 | - |
| CO2 | L3 | 3 | 3 | 2 | 2 | 2 | - | - | - | 1 | 2 | - |
| CO3 | L4 | 3 | 3 | 2 | 2 | 2 | - | - | - | - | 1 | - |
| CO4 | L4, L5 | 3 | 3 | 2 | 3 | 3 | - | - | - | 1 | 2 | - |
| CO5 | L5, L6 | 2 | 3 | 3 | 2 | 3 | 1 | - | 1 | 2 | 2 | 1 |

## Textbooks

1. Timoshenko S., Young D.H., Rao J.V. and Pati S., Engineering Mechanics, McGraw Hill Education, 5th Edition, 2017.
2. Bhavikatti S.S., A Textbook of Engineering Mechanics, New Age International Publications, 4th Edition, 2018.
3. Timoshenko S. and Gere J., Mechanics of Materials, CBS Publishers, 2006.
4. Strengthofmaterials,R.K.Rajput,S.Chand&Co,NewDelhi
5. Strength of Materials, Basavarajaiah B.S. and Mahadevappa P., Universities Press, 3rd Edition, 2010

## Reference Books

1. Hibbeler R.C., Engineering Mechanics: Statics and Dynamics, Pearson Education, 14th Edition, 2022.
2. Meriam J.L. and Kraige L.G., Engineering Mechanics (Statics & Dynamics), John Wiley, 6th Edition, 2008.
3. Shames I.H., Engineering Mechanics: Statics and Dynamics, PHI, 4th Edition, 2002.
4. Beer F.P. and Johnston E.R., Vector Mechanics for Engineers (Statics & Dynamics), McGraw Hill, 12th Edition, 2019.
5. Bhattacharya B., Introduction to Statics and Dynamics, Oxford University Press, 2nd Edition, 2014.
6. Goldberg D.E., Schaum's Outline of Fluid Mechanics and Hydraulics, McGraw Hill, 4th Edition, 2014.

### Web Resources & NPTEL Links

- NPTEL: Engineering Mechanics – https://nptel.ac.in/courses/112105171
- NPTEL: Rigid Body Dynamics – https://nptel.ac.in/courses/112104120
- PhET Simulations (University of Colorado, Free): https://phet.colorado.edu/en/simulations
- MIT OpenCourseWare – Classical Mechanics: https://ocw.mit.edu/courses/8-01sc-classical-mechanics
- Google Colab for Python: https://colab.research.google.com
- OpenSees documentation: https://opensees.berkeley.edu
- Tracker Video Analysis (Free): https://physlets.org/tracker

| Recommended Free & Open-Source AI/Computational Tools |
|---|
| Python (NumPy, SciPy, Matplotlib, scikit-learn) – Available FREE on Google Colab |
| Scilab 6.x – Free, open-source MATLAB alternative for numerical computation (scilab.org) |
| OpenSees – Free, open-source structural/earthquake engineering simulation framework |
| FreeCAD – Free, open-source 3D CAD with section property computation (freecadweb.org) |
| Tracker – Free video-based physics analysis tool for kinematics experiments (physlets.org/tracker) |
| GeoGebra – Free interactive mathematics and physics visualization (geogebra.org) |
| Algodoo – Free 2D physics simulation environment (algodoo.com) |

I Year – I Semester

| Course Code | DESIGN THINKING | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26ACS5101T | Common to All Branches | 1 | 0 | 4 | 0 | 3 |

## Course Overview

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

## Detailed Syllabus

### UNIT 1: Introduction to Design Thinking

**Core Concepts:**

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

**AI Integration:**

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

### UNIT 2: Empathise and Define Stages

**Core Concepts:**

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

**AI Integration:**

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

## UNIT 3: Ideation Stage

### Core Concepts:

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

### AI Integration:

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

## UNIT 4: Prototyping

### Core Concepts:

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

### AI Integration:

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

## UNIT 5: Testing, Iteration and AI in Design

### Core Concepts:

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

### AI Integration:

- AI Tool: Maze (free plan) – conduct remote usability tests on Figma prototypes
- AI Tool: 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

| 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 Mapping Matrix

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

| CO/PO | PO 1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO1 0 | PO1 1 | PO1 2 | PSO 1 | PSO 2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 2 | 1 | 1 | 2 | 1 | 1 | 2 | 2 | 1 | 2 | 1 | 1 |
| CO2 | 1 | 2 | 3 | 2 | 2 | 3 | 1 | 1 | 3 | 2 | 2 | 2 | 2 | 2 |
| CO3 | 2 | 3 | 3 | 2 | 3 | 3 | 1 | 1 | 2 | 2 | 2 | 3 | 2 | 2 |
| CO4 | 2 | 3 | 3 | 3 | 3 | 2 | 1 | 2 | 2 | 2 | 2 | 3 | 2 | 2 |
| CO5 | 2 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 |
| Averag e | 1.6 | 2.6 | 2.8 | 2.2 | 2.4 | 2.4 | 1.2 | 1.4 | 2.4 | 2.2 | 2.0 | 2.6 | 2.0 | 2.0 |

### Text Books


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


### Reference Books


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

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

### Web Resources

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

### YouTube / Video Resources

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

| Course Code | COMPUTATIONAL THINKING AND PROBLEM SOLVING WITH PYTHON | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26ACS5102T | Common to All Branches | 3 | 0 | 0 | 0 | 3 |

**I Year I Sem**

## Course Objectives

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

## Course Outcomes

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

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

### UNIT – I :COMPUTATIONAL THINKING AND PROGRAMMING BASICS

- Computational thinking: characteristics, problem-solving strategies, steps in problem solving; algorithms — definition and properties; flowcharts — symbols and construction; pseudocode — writing and conversion; abstraction, decomposition, pattern recognition, algorithm efficiency basics.
- Introduction to Python: installation and execution environment; variables, identifiers, keywords; data types, type conversion; input and output statements; expressions and operators — arithmetic, relational, logical, assignment; operator precedence.

**Source:** TB1: Chapters 1–3 (Spronck — computational thinking, algorithms, flowcharts, Python basics); TB2: Chapters 1–2 (Thareja — introduction to programming, Python fundamentals)

**AI Tools Integration:** Use Flowgorithm to design and simulate algorithms (e.g., largest of 3 numbers); draw flowcharts using Lucidchart; use Algorithm Visualizer to trace algorithm execution; use ChatGPT to convert pseudocode to Python and verify output.

**Unit 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

- Decision control statements: Boolean expressions; if, if-else, if-elif-else, nested if; conditional expressions (ternary operator).

- Looping statements: while loop, for loop, iteration techniques, nested loops, infinite loops; loop control — break, continue, pass; else with loops.
- Practical problem solving: prime number check, pattern programs using nested loops, menu-driven programs using control statements.

**Source:** TB2: Chapters 4–5 (Thareja — decision making, loops, control statements); TB1: Chapters 4–5 (Spronck — conditions, loops, iteration)

**AI Tools Integration:** Use Python Tutor to visualize if-else and loop execution step by step; use Thonny to debug loop-based programs; use Replit for online coding practice; use ChatGPT to generate test cases for control flow programs.

**Unit 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

- Strings: representation, indexing, slicing, operations, built-in functions and methods.
- Lists: creation, indexing and slicing, operations, functions, methods, nested lists.
- Tuples: creation, operations, packing and unpacking.
- Sets: creation, set operations — union, intersection, difference; frozen sets.

**Source:** TB2: Chapters 6–8 (Thareja — strings, lists, tuples, sets, dictionaries); TB1: Chapters 6–7 (Spronck — strings, lists, tuples)

**AI Tools Integration:** Use NLTK and TextBlob to perform text analysis (sentiment, word frequency) on strings; use Pandas and NumPy for list and array operations; use ChatGPT to explain string slicing and list comprehensions with concrete examples.

**Unit 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

- Dictionaries: creation, operations, methods; dictionary-based applications.
- Functions: built-in and user-defined functions; function definition and calling; arguments — positional, keyword, default, variable-length; scope of variables (local and global).
- Recursion: recursive functions — factorial, Fibonacci; lambda functions (anonymous functions); applications of functions in problem solving.

**Source:** TB2: Chapters 9–10 (Thareja — functions, recursion, lambda, dictionaries); TB1: Chapters 8–9 (Spronck — functions, recursion, problem solving)

**AI Tools Integration:** Use Jupyter Notebook for interactive function experiments; use SymPy for symbolic mathematical expressions; use PyCharm for modular program development; use ChatGPT to trace recursive function calls step by step.

**Unit 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

- Modules and packages: creating and importing modules; standard library modules.
- File handling: opening, reading, writing, and closing files; file modes; working with text files; processing CSV and Excel files.
- Exception handling: types of errors (syntax, runtime, logical); try, except, finally blocks; raising exceptions.

- Introduction to OOP: classes, objects, attributes, methods, constructors, self keyword; basic applications of OOP in Python.

**Source:** TB2: Chapters 11–13 (Thareja — modules, file handling, exception handling, OOP); TB1: Chapter 10 (Spronck — file handling, exceptions, object-oriented programming)

**AI Tools 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.

**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.

## LEARNING RESOURCES

### Textbooks

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

### Reference Books

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

### Online Tools Used in This Course

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

| Course Code | AI TOOLS AND APPLICATIONS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26ACS5103T | Common to All Branches | 1 | 0 | 4 | 0 | 3 |

I Year I Sem

## 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

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

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

## UNIT – I : UNDERSTANDING AI AND AI FOR DOCUMENTS AND PRESENTATIONS

- 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 Tools 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 Outcomes**

- Explain AI concepts and the AI tool landscape to a non-technical audience.
- Generate and critically refine presentations using Gamma.app, Copilot, and Canva AI.
- Use AI writing tools for drafting, editing, and improving academic documents.

## UNIT – II :AI FOR RESEARCH, NOTE-TAKING, DATA, AND SPREADSHEETS

- 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 Tools 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 Outcomes**

- Use NotebookLM to generate source-grounded study aids from uploaded documents.
- Apply AI research tools (Perplexity, Gemini Deep Research, Elicit) and verify citations.
- Perform spreadsheet automation and data analysis using AI; use Wolfram Alpha, Symbolab, and Photomath to solve and verify equations.

## UNIT – III :AI FOR VISUAL DESIGN, DIAGRAMS, AND MEDIA

- 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 Tools 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

- 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 Tools 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

- 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 Tools 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.

**Unit Outcomes**

- Explain AI bias, privacy risks, and governance frameworks including India's DPDP Act 2023.
- Apply a personal data safety protocol when using public AI tools.
- Design, execute, and present an AI-assisted mini-project with responsible usage disclosure.

## LEARNING RESOURCES

### Reference Books

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

### Free Online Courses

- Google – Fundamentals of AI – grow.google/intl/en_in/ai-learning-resources

- Microsoft – AI Skills Navigator – microsoft.com/en-us/ai/ai-skills
- Andrew Ng – AI for Everyone (Coursera, free to audit) – coursera.org/learn/ai-for-everyone
- Elements of AI – University of Helsinki (free, beginner) – elementsofai.com

### Tools Used in This Course (All Free-Tier Accessible)

- AI Assistants: ChatGPT – chat.openai.com | Claude – claude.ai | Gemini – gemini.google.com | Microsoft Copilot – copilot.microsoft.com
- Research & Notes: NotebookLM – notebooklm.google.com | Perplexity AI – perplexity.ai | Elicit – elicit.org | Consensus – consensus.app
- Presentations & Docs: Gamma.app | Notion AI – notion.so |Beautiful.ai | Copilot in Office 365 | Gemini in Google Workspace
- Diagrams & Design: Napkin.ai | Whimsical AI – whimsical.com | Miro AI – miro.com | Canva AI – canva.com | DALL-E 3 via ChatGPT
- Data & Spreadsheets: ChatGPT Advanced Data Analysis | Julius AI – julius.ai | Copilot in Excel | Gemini in Sheets
- 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
- Audio & Video: ElevenLabs – elevenlabs.io | Suno – suno.com | Runway – runwayml.com |HeyGen – heygen.com
- Detection & Automation: GPTZero – gptzero.me |Originality.ai | Google Lens |Make.com | GitHub Copilot

### Web References

- 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 |
|---|---|---|---|---|---|---|
| 26AME3101P | Common to All Branches | 0 | 0 | 0 | 3 | 1.5 |

**Prerequisites:**Basic Computing Elements

**Course objectives:** The objectives of the course are to

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

**Course Outcomes:** On successful completion of the course, the student will be able to,

| | |
|---|---|
| 1 | Recognize the importance of safety precautions while working in the workshop. |
| 2 | Identify different measuring, marking, cutting tools used in the workshop and their operational capabilities. |
| 3 | Fabricate precision wood and metal fitting joints within standard specifications. |
| 4 | Understand Wire standard, safe residential switches, lighting circuits, and basic appliances. |
| 5 | Identify different tools used in foundry and welding sections |

**1. Safety Practices & Precautions**
**Demonstrate the necessity of** Workshop hazards, various Personal Protective Equipment and emergency protocols.
**AI Integration:** Computer vision for safety compliance.
**2. Wood working:**Introduction to different types of woods, carpentry tools, timber joints

a) Half – Lap joint b) Mortise and Ten on joint c) Corner Dovetail joint or Bridle joint
**3. Fitting:** 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:**Familiarity with different types of tools used in sheet metal working, Developments of following sheet metal job from GI sheets.
a) Tapered tray b) Conical funnel c) Elbow pipe d) Brazing
**5. Electrical wiring:**Introduction to electrical tools, safety precautions, wiring materials. Familiarity with different types of basic electrical circuits and make the following connections.

a) Parallel and series b) Two-way switch c) God own lighting d) Tube light e) Three phase motor f) Soldering of wires

**6. Plumbing:** 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**: Demonstration and practice on Arc Welding and Gas welding. Preparation of a simple Lap joint/Butt joint.

**8. Foundry Trade:** Demonstration and practice on Moulding tools and processes, Preparation of Green Sand Moulds for a simple Patterns.

8. **Introduction to 3D Printing:** Demonstrate the core concepts of 3D printing, Hardware, software, Digital file and 3D printing materials.Familiarise with 3D Printing Process of complex objects.
9. **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 the procedures to resolving them.

**Peripheral Installation:**

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.

10. **Disassembly and Assembly of equipment such as Bicycle, Gear Box, Braking System: F**amiliarise with disassembly and assembly process

**AI Integration Module**: 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.

## Learning Resources

## Textbooks

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

## Reference Books

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

## Video Links & Online Lectures

- **Classical Shopfloor Operations**: NPTEL Workshop Practices Video Series by IIT Kharagpur
- **Industrial AI & Predictive Analytics**: Predictive Maintenance Models in Python Tutorial

- **Smart Sensors & Electronics Fabrication**: MIT 6.131: Power Electronics and Smart Grid Interfaces

**Engineering Workshop**

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

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

I Year I Sem

# MECHANICAL ENGINEERING CHEMISTRY LAB (AI INTEGRATED)

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0104P | I Year I Sem | 0 | 0 | 0 | 3 | 1.5 |

**Branch:** Mechanical Engineering (ME) | Regulation: JNTUA R26 (per source file name; not stated as a labelled field in the source document.)

## Course Objectives

*(Suggested — Course Objectives were not stated in the source; derived from the experiment list below.)*

- Perform redox, acid-base, and instrumental titrations to determine the concentration and strength of chemical species.
- Analyze coal samples through proximate analysis to determine moisture content.
- Verify adsorption isotherms and apply them to understand solid-liquid interface phenomena.
- Study the corrosion behavior of metals under different environmental conditions.
- Apply instrumental methods (conductometry, potentiometry, pH-metry, spectrophotometry, viscometry) for quantitative chemical analysis relevant to mechanical engineering.

## List of Experiments

### GENERAL CHEMISTRY EXPERIMENTS

- 1.Estimation of Ferrous Iron (Fe+2) using potassium dichromate (K2Cr2O7) (Redox titrations)
- 2.Determination of strength of acid (HCl) using Sodium Hydroxide
- 3.Determination of moisture content in coal sample (Proximate analysis)
- 4.Verification of Freundlich Adsorption isotherm (Adsorption of acetic acid by charcoal)
- 5. Determination of corrosion.
- 6. Determination of corrosion by environment.

### Instrumental methods of chemical analysis

- 7. Conductometric titrations of strong acid Vs strong base (Ex: HCl Vs NaOH)
- 8. Potentiometry (Redox titrations)-Estimation of Iron (Fe+2) using potassium dichromate (K2Cr2O7)
- 9. pH metry - Estimation of Hydrochloric acid (HCl) using Sodium Hydroxide (NaOH)
- 10. Spectrophotometry - Determination of wavelength (λ) and verification of Beer's – Lamberts law.
- 11. Determination of viscosity of lubricating oil using Red-Wood Viscometer - I
- 12. Determination of viscosity of lubricating oil using Red-Wood Viscometer - II

**AI Extension:** Data acquisition and analysis using Python/Excel. Predictive analysis for corrosion and lubrication behavior.

## Course Outcomes (COs)

*(Suggested — Course Outcomes were not stated in the source; derived from the experiment list.)*

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Estimate the concentration of ferrous iron and acid strength using redox and acid-base titrations. | L3 (Apply) |
| CO2 | Determine moisture content in coal samples and verify the Freundlich adsorption isotherm. | L3 (Apply) |
| CO3 | Analyze corrosion behavior of metals under different environmental conditions. | L4 (Analyze) |
| CO4 | Perform conductometric, potentiometric, and pH-metric titrations using instrumental methods. | L3 (Apply) |
| CO5 | Determine wavelength using spectrophotometry and viscosity of lubricating oils using a Red-Wood Viscometer. | L3 (Apply) |

## CO – PO Articulation Matrix

*(Suggested — a formal CO–PO mapping was not provided in the source.)*

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L3 | 3 | 2 | - | 1 | 2 | - | - | - | - | - | - | 1 |
| CO2 | L3 | 3 | 2 | 1 | 1 | 2 | - | 1 | - | - | - | - | 1 |
| CO3 | L4 | 3 | 3 | 1 | 2 | 2 | - | 1 | - | - | - | - | 2 |
| CO4 | L3 | 3 | 2 | 2 | 1 | 3 | - | - | - | - | - | - | 1 |
| CO5 | L3 | 2 | 2 | 2 | 1 | 3 | - | - | - | - | - | - | 1 |

## Online Learning Resources

- STLE
- ASM International
- SAE
- ASME
- NACE

## Recommended AI Tools

**Python:** NumPy, Pandas, Matplotlib

**ML Frameworks:** TensorFlow / Scikit-learn

**Computer Vision:** OpenCV

**Data Mining:** Orange Data Mining

**Life-Cycle Analysis:** OpenLCA

| Course Code | COMPUTATIONAL THINKING AND PROBLEM SOLVING WITH PYTHON LAB | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26ACS5102P | Common to All Branches | 0 | 0 | 0 | 3 | 1.5 |

**I Year I Sem**

## 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 ($\pi$ $r^2$) and circumference ($2\pi$ 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.

## LEARNING RESOURCES

### Textbooks

- TB1: Pieter Spronck, Computational Thinking with Python (course textbook — algorithms and Python foundations).
- TB2: Reema Thareja, Programming in Python, Oxford University Press. (Primary implementation reference — all five units.)

### Reference Books

- Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, O'Reilly. (Free at greenteapress.com)
- 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 Semester

| Course Code | NSS / NCC / Scouts & Guides / Community Service | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0101L | Common to All Branches | 0 | 0 | 0 | 1 | 0.5 |

## Course Objectives

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

## Course Outcomes (COs)

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

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

## Unit-wise Syllabus with Learning Outcomes

| UNIT 1 | Foundation of NSS, NCC, Scouts & Guides and Community Service | 5 Hrs |
|---|---|---|
| Topics | • Introduction to NSS – history, objectives, badge, motto, pledge<br>• Introduction to NCC – ranks, uniforms, aims, NCC Act 1948<br>• Scouts & Guides – Baden-Powell legacy, Bharat Scouts & Guides Act<br>• Community Service – concept, importance and types<br>• Citizenship, civic sense and social responsibility<br>• Volunteerism: global and Indian perspective (NYKS, NSS, Red Cross)<br>• Legal & ethical framework for community service | |
| Unit-wise Learning Outcomes (ULOs) | **On completion of this unit, students will be able to:**<br>• ULO1.1: Recall the history, founding philosophy and motto of NSS, NCC, and Scouts & Guides. [Remember]<br>• ULO1.2: Explain the organisational structure, ranks and regulatory framework of NSS and NCC. [Understand] | |

| | |
|---|---|
| | • ULO1.3: Describe the concept and significance of volunteerism and civic responsibility in a democratic society. [Understand]<br>• ULO1.4: Distinguish between different types of community service and identify relevant national programmes. [Analyse] |

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

| UNIT 3 | Community Projects, Social Awareness and Service Documentation | 5 Hrs |
|---|---|---|
| **Topics** | • Community needs assessment – participatory methodology<br>• Project planning: Swachh Bharat, Pulse Polio, Blood Donation camps<br>• Literacy campaigns and awareness drives (PadhnaLikhna Abhiyan)<br>• Environmental conservation – plantation, solid waste, water conservation<br>• Gender sensitisation and women empowerment programmes<br>• Digital literacy and e-governance outreach<br>• Documentation and reporting of service-learning activities<br>• Reflection, portfolio preparation and field report writing | |
| **Unit-wise Learning Outcomes (ULOs)** | **On completion of this unit, students will be able to:**<br>• ULO3.1: Conduct a community needs assessment using participatory tools. [Analyse]<br>• ULO3.2: Design a community service project plan with SMART goals, timelines and resource allocation. [Create]<br>• ULO3.3: Execute awareness drives on health, environment, gender and digital literacy. [Apply]<br>• ULO3.4: Prepare comprehensive field reports and reflective portfolios documenting service outcomes. [Create]<br>• ULO3.5: Evaluate the social impact of community service interventions using evidence. [Evaluate] | |

## Text Books

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

## Reference Books

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

## Web Resources

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

## YouTube Resources

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

## AI-Integrated Activities

| S.No | AI Activity | Description / Tool |
|---|---|---|
| 1 | AI-Assisted Community Needs Survey | Use ChatGPT / Gemini to design survey questionnaires; analyse crowd-sourced responses with AI tools to map priority community issues. |

| | | |
|---|---|---|
| 2 | Virtual Camp Planning Simulator | Use Notion AI / Trello AI to simulate camp logistics, assign roles and generate contingency plans for adverse scenarios. |
| 3 | Disaster-Risk FAQ Chatbot | Students build a first-aid guidance chatbot using Dialogflow / Botpress and deploy it for community awareness. |
| 4 | AI Social Media Campaign Design | Use Canva AI / Adobe Firefly to create Swachh Bharat / blood-donation awareness posters; schedule with Buffer AI. |
| 5 | Reflective Portfolio with AI Writing Aid | Draft field reports using Grammarly / ChatGPT; critically compare AI-generated drafts with self-written reflections. |

## CO – PO / PSO Mapping

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

| CO / PO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | 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 | - | - | - | - | - |

**I Year II Sem**

| Course Code | COMMUNICATIVE ENGLISH | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0201T | Common to All Branches | 2 | 0 | 0 | 0 | 2 |

*Prerequisites: Basic English (Intermediate level)*

## Course Objectives

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

## Course Content

## UNIT I Foundations of English Communication

### 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:

Grammar explanation, error identification, and instant Q&A on language rules, Real-time grammar, spelling, and punctuation correction with detailed explanations, Phonetics practice – voice recording for pronunciation and intonation feedback, Word-of-the-day, etymology, and interactive vocabulary quizzes.

### Free/Open-Source AI Tools:

Grammarly (free); ChatGPT / Gemini (free tiers); Google Speech-to-Text (free); Merriam-Webster Vocabulary Builder (free)

## UNIT II Listening and Speaking Skills

### 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:

AI-powered presentation design – structured, visual, professional slides, Fluency analysis, filler-word detection, pacing, and confidence scoring, Auto-transcription and keyword analysis of spoken communication, Visual communication design – infographics, posters, and slide aesthetics

**Free/Open-Source AI Tools:**

Gamma.app (free); Orai AI Speech Coach (free); Otter.ai (free tier); Canva AI (free)

| UNIT III | Reading Comprehension & Writing Skills |
|---|---|

**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 and blogs.

**AI Integration:**

AI paraphrasing tool – rewrite passages at varying complexity levels, Readability scoring, passive voice detection, and sentence simplification, Essay outline generation, argument structuring, and content brainstorming, Vocabulary simplification for improving comprehension of complex texts

**Free/Open-Source AI Tools:**

QuillBot (free); Hemingway Editor (free online); ChatGPT (free); Rewordify (free)

| UNIT IV | Professional & Business Communication |
|---|---|

**Core Content:**

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

**AI Integration:**

Draft and refine professional emails, cover letters, reports, and proposals, AI-powered resume builder with ATS-optimized templates, Report structuring, agenda writing, and knowledge management, Tone detection, formality check, and professional register evaluation

**Free/Open-Source AI Tools:**

ChatGPT (free); Kickresume (free tier); Grammarly (free); Notion AI (free tier)

| UNIT V | Use of AI in Language Learning |
|---|---|

**Core Content:**

Artificial Intelligence in Communication: meaning, basics, benefits and limitations of AI in human interaction. AI chatbots and virtual assistants in education: grammar correction, vocabulary building, AI-based speaking and pronunciation tools, writing assistants, e-portfolio creation. AI in Global Communication: automated translation, summarizing large texts, finding information in different languages. Digital Communication and AI Ethics: responsible use of AI in education, plagiarism and academic honesty, fake information and AI-generated content, privacy, bias, and ethical communication online.

**AI Integration:**

Plagiarism detection, paraphrase checker, and academic integrity analysis, High-accuracy translation supporting intercultural communication understanding, AI literacy modules – understanding AI bias, ethics, and responsible usage, E-portfolio creation platform for showcasing communication artifacts

### Free/Open-Source AI Tools:

Copyleaks / Turnitin (institutional access); DeepL Translator (free); Wix (free); Adobe Portfolio (free for students); ChatGPT / Gemini (free tiers)

## Course Outcomes (COs)

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

## 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 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | L1,L2 | 1 | – | – | 1 | – | 1 | – | 1 | 1 | 3 | 1 | 2 | – | – |
| **CO2** | L2 | 1 | – | – | 1 | 1 | 2 | 1 | 1 | 2 | 3 | 1 | 2 | – | – |
| **CO3** | L3 | – | – | 1 | – | 3 | 2 | 1 | 2 | 3 | 3 | 2 | 2 | – | 1 |
| **CO4** | L4 | – | 1 | 1 | 2 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 2 | – | 1 |
| **CO5** | L5 | – | 1 | 1 | 2 | 3 | 2 | 2 | 3 | 2 | 3 | 2 | 3 | – | 1 |
| **CO6** | L6 | – | 1 | 2 | 2 | 3 | 3 | 2 | 3 | 3 | 3 | 3 | 3 | – | 2 |

## Textbooks

1. Raman M. and Sharma S., Technical Communication: Principles and Practice, Oxford University Press, 4th Edition, 2022.

2. Shetty J. and Pareek R., Communicative English for Engineers and Professionals, Wiley India, 3rd Edition, 2021.

## Reference Books

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

## Web Resources & NPTEL Links

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

**I Year II Sem**

# PHYSICS FOR MECHANICAL ENGINEERS

## (AI Integrated)

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0201T | I Year I Sem | 3 | 0 | 0 | 0 | 3 |

**Prerequisites:** Engineering Mechanics, Engineering Mathematics, Basic Physics

## Course Objectives

The objective of this course is to:

1. To provide students with a strong foundation in wave optics concepts such as interference, diffraction and polarization along with their applications in optical measurements, dimensional
inspection and basic AI-assisted image and fringe pattern analysis techniques.
2. To introduce the fundamentals of crystallography and X-ray diffraction methods for understanding crystal structures, phase identification and material characterization using simple
AI-assisted data analysis approaches.
3. To equip students the basic principles of quantum mechanics and quantum computing, including quantum states, tunneling, superposition and secure communication with introductory
AI-assisted quantum simulations and visualization tools.
4. To familiarize students with oscillations, vibration control systems, lasers and precision engineering applications along with basic AI-assisted monitoring, predictive maintenance and intelligent control concepts used in modern industries.
5. To impart knowledge on the principles of ultrasonics and non-destructive testing methods for
defect detection, quality inspection and structural analysis using simple AI-assisted signal interpretation and automated inspection techniques.

## Course Content

### UNIT I: Applied Wave Optics
9 hrs

**Core Content:**

Interference: Principle of superposition and coherence, Interference, Types of interference, Interference in thin films (reflection geometry) and Newton's rings. Applications: Determination of wavelength and refractive index.

Diffraction and Optical Surveying: 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: 3D Scanning for Dimensional Inspection of Casted Parts.

Polarization: Types of polarization, Nicol's prism, Half-wave and Quarter-wave plates.

**AI Integration:**

AI-integrated holographic interferometry for micro-deformation analysis. Python-based fringe pattern analysis using image processing (OpenCV).

**Free/Open-Source AI Tools:**

Python (NumPy, SciPy, Matplotlib and scikit-learn).

## UNIT II: Crystallography and AI-Driven Material Engineering 8 hrs

**Core Content:**

Crystallography: Space lattice, Basis, Unit Cell and lattice parameters – Crystal systems, Bravais Lattices — Coordination number - Packing fraction of SC, BCC & FCC - Miller indices – Separation between successive (h k l) planes.

X-ray diffraction: Bragg's law - X-ray Diffractometer – Crystal structure determination by Laue's method. Application:XRD Pattern Analysis for Phase Identification, Determination of Crystal Structure of Metals and Alloys

**AI Integration:**

AI-assisted XRD pattern analysis for phase identification and crystal structure determination. ML-based classification of diffraction patterns using scikit-learn.

**Free/Open-Source AI Tools:**

Python (NumPy, SciPy, Matplotlib 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, Schrödinger's time-independent wave equation, Particle in a one-dimensional potential well, Quantum tunnelling (basic idea).

Introduction to Quantum Computing: Classical bits vs qubits. Concept of superposition and Entanglement, Basic Quantum Operations: Hadamard gate (idea of superposition), Pauli-X gate. Basic idea of quantum measurement,Applications: Basic idea of quantum cryptography and secure communication.

**AI Integration:** AI-assisted quantum circuit simulation using IBM Qiskit (free). Python-based visualization of wave functions and quantum probability densities

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

## UNIT IV: Oscillations, Vibration Control and Laser-Based Metrology 10 hrs

**Core Content:**

Introduction to oscillations and oscillations of springs. Series and parallel combinations of springs in Suspension Systems, Stiffness factor and its physical significance, vibration isolation. Theory of damped oscillations (Over, Under and Critical damping – all qualitative).

Forced oscillations (qualitative) and Resonance; Sharpness of resonance and Quality factor (Q). Applications (Qualitative): shock absorbers, noise, and vibration reduction.

Lasers and Precision Engineering: Principles of Absorption, Spontaneous and Stimulated Emission,Population Inversion. Three-level and Four-level laser systems. Construction and working of He-Ne Laser and Nd-YAG Laser. Applications (qualitative only): Role of high-power lasers in CNC Laser Cutting and Welding.

**AI Integration:** AI-Assisted Adaptive Suspension System for EVs, AI-Based Vibration Monitoring and Predictive Maintenance.

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

## UNIT V: Ultrasonics and Non-Destructive Testing (NDT) 8 hrs

**Core Content:**

Introduction to ultrasonics; Production of ultrasonics: Piezoelectric method and Magnetostriction method, Detection of ultrasonics: Kundt's tube and Quartz crystal methods. Applications: Use of Ultrasonic Flaw Detectors to identify internal cracks, voids and thickness variations in cast and forged engine components.

**Non-Destructive Testing (NDT):** Introduction, need and advantages in mechanical engineering structures, Basic inspection methods: Visual inspection and AI-based drone inspections for inaccessible materials (qualitative). Liquid penetrant testing: basic principle for detecting surface defects in steel. Applications: Detecting surface and sub-surface defects, sorting alloys and measuring coating thickness on automotive parts.

**AI Integration:** AI-recognition in abnormal echo patterns,AI-assistedadditional reflections and frequency distortions.

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

**Course Outcomes (COs)**

After completion of the course, student will be able to

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| **CO1** | Examine the principles of interference, diffraction and polarization and utilise basic AI-assisted optical analysis techniques in engineering applications. | L3 (Apply) |
| **CO2** | Analyze crystal structures, diffraction patterns and material properties using X-ray diffraction methods and basic AI-assisted material characterization tools. | L4 (Analyze) |
| **CO3** | Interpret the concepts of quantum mechanics and quantum computing and apply basic AI-assisted quantum simulation techniques for secure communication systems. | L3 (Apply) |
| **CO4** | Analyze oscillatory systems, vibration control mechanisms and laser-based metrology techniques using basic AI-assisted monitoring and predictive | L4 (Analyze) |

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| | approaches. | |
| **CO5** | Evaluate ultrasonic testing and non-destructive testing methods for defect identification and quality assessment using basic AI-assisted signal analysis techniques | 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 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 3 | 1 | 2 | 3 | - | - | - | - | 1 | - | 2 |
| CO2 | 3 | 3 | 2 | 2 | 3 | - | - | - | - | 1 | - | 2 |
| CO3 | 3 | 2 | 2 | 2 | 3 | - | - | - | - | 1 | - | 2 |
| CO4 | 3 | 3 | 2 | 2 | 3 | - | - | - | - | 1 | 1 | 2 |
| CO5 | 3 | 3 | 2 | 3 | 3 | 1 | - | - | - | 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

3. B. K. Pandey & S. Chaturvedi, Engineering Physics, Cengage Learning, 2021.
4. Shatendra Sharma &Jyotsna Sharma, Engineering Physics, Pearson, 2018.
5. Sanjay D. Jain, D. Sahasrabudhe & Girish, Engineering Physics, Universities Press, 2010.
6. M. R. Srinivasan, Engineering Physics, New Age International, 2009.
7. S. O. Pillai, Solid State Physics, New Age International.
8. R. Murugeshan & Kiruthiga Sivaprasath, Modern Physics, S. Chand.

## 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

## Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for wave optics and signal analysis
- TensorFlow / Keras (free, open-source) – For crack and defect detection CNNs on Google Colab
- scikit-learn (free, open-source) – For anomaly detection in vibration and NDT data
- IBM Qiskit (free, open-source) – For quantum computing simulations on Google Colab
- OpenCV (free, open-source) – For image processing in visual inspection and interferometry
- PhET Interactive Simulations (free) – For wave optics, quantum mechanics and oscillations

**I Year II Sem**

| Course Code | DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0203T | Common to All Branches | 3 | 0 | 2 | 0 | 4 |

*Prerequisites: Linear Algebra and Calculus*

## Course Objectives

- Develop analytical and computational competence in solving ordinary and partial differential equations arising in engineering systems.
- Enable students to interpret vector differential and integral operators in physical and engineering contexts.
- Integrate mathematical modeling, visualization, and simulation using modern AI-assisted computational tools in alignment with NEP 2020.
- 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 SymPydsolve 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-factorinterpretation.

**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 and functions; first-order linear PDEs and Lagrange's method; 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; Wolfram Alpha 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 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.

### 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 | Analyse 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. | 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 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L3 | 3 | 3 | 2 | 2 | 1 | – | – | – | 1 | 2 | – | – | 2 | 1 |
| CO2 | L4 | 3 | 3 | 3 | 2 | 2 | – | – | – | 1 | 2 | – | – | 2 | 1 |
| CO3 | L3 | 3 | 2 | 2 | 2 | 1 | – | – | – | – | 2 | – | – | 2 | 1 |
| CO4 | L4 | 3 | 3 | 3 | 3 | 2 | – | – | – | 1 | 2 | – | – | 2 | 2 |
| CO5 | L5 | 3 | 3 | 2 | 3 | 3 | – | 1 | 1 | 2 | 2 | 1 | – | 2 | 2 |

## Textbooks

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

## Reference Books

1. Zill D.G. and Wright W.S., Advanced Engineering Mathematics, 6th Edition, Jones & Bartlett, 2018.
2. Sneddon I.N., Elements of Partial Differential Equations, Dover Publications, 2006.

3. Arfken G.B. et al., Mathematical Methods for Physicists, 7th Edition, Academic Press, 2012.

## Web Resources & NPTEL Links

- NPTEL: ODEs, PDEs, and Vector Calculus – https://nptel.ac.in
- MIT OpenCourseWare – Differential Equations (18.03) and Multivariable Calculus (18.02): https://ocw.mit.edu
- Python SymPy Documentation: https://docs.sympy.org
- SciPy Documentation: https://docs.scipy.org

**I Year II Sem**

| Course Code | ENGINEERING MECHANICS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3202T | **Professional Core-1**<br>Common to Civil & Mechanical Branches | 3 | 0 | 0 | 0 | 3 |

**Prerequisites:** Engineering Mathematics (Linear Algebra & Calculus, Differential Equations)

## Course Objectives

- Introduce fundamental concepts of mechanics, force systems, and equilibrium with applications in structural and mechanical engineering.
- Develop ability to draw and interpret free body diagrams for coplanar, concurrent, and spatial force systems.
- Calculate centroid, centre of gravity, and moment of inertia for simple and composite bodies.
- Understand the fundamental concepts of stress, strain, elasticity, and the mechanical behaviour of materials under different loading conditions.
- Introduce AI/ML tools for simulation, analysis, and visualization of mechanical systems.

## Course Content

### UNIT I: Introduction, Force Systems & Friction

**Core Content:** Basic concepts of mechanics: scalar and vector quantities, Newton's laws, SI units. Coplanar concurrent forces: resultant by parallelogram law, triangle law, polygon law. Moment of a force: Varignon's theorem. Couples and resultants of force systems. Lami's theorem. Spatial force systems: vector representation, dot and cross products. Friction: limiting friction, Coulomb's laws of dry friction, coefficient of friction, angle of friction, cone of static friction, impending motion problems, wedge friction, screw jack.

**AI Integration:** Introduction to AI in Engineering: concept of data, machine learning (ML), and simulation. Using Python (NumPy) to solve force equilibrium problems computationally. Introduction to MATLAB/Scilab for vector force calculations. Visualizing force systems using Matplotlib/Python.

**Free/Open-Source AI Tools:** Python (NumPy, Matplotlib) – Google Colab (free); Scilab (free MATLAB alternative); GeoGebra (free online vector visualization)

### UNIT II: Equilibrium of Force Systems & Virtual Work

**Core Content:** Equilibrium conditions for coplanar systems: free body diagrams, equations of equilibrium ($\Sigma Fx=0$, $\Sigma Fy=0$, $\Sigma M=0$). Graphical methods: triangle and polygon of forces. Equilibrium of spatial systems. Analysis of plane trusses: method of joints, method of sections. Analysis of frames. Principle of virtual work: statement, applications to simple mechanisms and trusses.

**AI Integration:** Using Python/OpenSees (open source) for automated truss analysis. AI-based structural analysis: introduction to how FEM software uses matrix methods internally. Computer simulation of mechanism motion using Python. Visual debugging of FBDs using interactive tools.

**Free/Open-Source AI Tools:** OpenSees (free, open-source structural analysis); Python scipy.optimize for equilibrium solution; SAP2000 student edition (free); online truss solvers (free)

### UNIT III: Centroid, Centre of Gravity & Moment of Inertia

**Core Content:** Centroid of lines, areas, and volumes from first principles. Centroid of standard shapes: rectangle, triangle, circle, semicircle, quarter circle. Centroid of composite figures using tabular method. Centre of gravity of 3D bodies: prismatic bodies, cones, hemispheres, composite bodies. Pappus theorems and their applications. Area moment of inertia: polar moment, transfer theorem (parallel axis theorem). MOI of composite sections. Product of inertia. Mass moment of inertia for simple and composite bodies.

**AI Integration:** Python scripting for automated centroid and MOI calculations of composite cross-sections. Using spreadsheets (LibreOffice Calc) and Python Pandas for tabular calculations.

Introduction to how CAD software (AutoCAD/FreeCAD free) computes section properties. Application in AI-driven structural design optimization: cross-section selection.

**Free/Open-Source AI Tools:** Python (NumPy, Pandas) – Google Colab; FreeCAD (free, open-source CAD with section property computation); LibreOffice Calc (free)

### UNIT IV: Kinematics & Kinetics of Particles

**Core Content:** Kinematics: rectilinear motion – displacement, velocity, acceleration relations; curvilinear motion – Cartesian and polar coordinates; projectile motion. Kinetics: Newton's second law; D'Alembert's principle; work-energy method: kinetic energy, work done, power, efficiency; impulse-momentum method: linear and angular; impact: direct and oblique central impact, coefficient of restitution.

**AI Integration:** AI in Autonomous Vehicles: how kinematic equations underpin self-driving car algorithms. Python simulation of projectile motion with drag (real-world). ML-based trajectory prediction: introduction to LSTM for motion prediction. Robot path planning using kinematic constraints. Using Python (matplotlib.animation) for interactive motion visualization.

**Free/Open-Source AI Tools:** Python (NumPy, Matplotlib, SciPy) – Google Colab; Tracker (free video analysis software for kinematics experiments); Algodoo (free 2D physics simulation)

### UNIT V: Simple Stresses, Strains & Elastic Constants

**Core Content:** Elasticity and plasticity; types of stresses (tensile, compressive, shear) and strains (linear, lateral, shear, volumetric); Hooke's law; Poisson's ratio; elastic constants: E, G, K and their relationships; factor of safety; bars of varying cross-section; elongation under self-weight; bars in series and parallel; temperature stresses; stresses in composite bars. Introduction to principal stresses and strains: principal planes, Mohr's circle for 2D stress state.**StrainEnergy**Resilience–Gradual,sudden,impactandshockloadings–simpleapplications.

**AI Integration:** ML for material property prediction: using regression models to predict Young's modulus and Poisson's ratio from material composition data. Python-based stress-strain curve fitting using SciPy. Introduction to data-driven material models in structural analysis. Overview of physics-informed neural networks (PINNs) for material mechanics (conceptual).

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, Matplotlib) on Google Colab; Scilab (free); MIT OpenCourseWare stress analysis tools

### Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| **CO1** | Remember and explain the fundamental concepts of mechanics including Newton's laws, types of forces, equilibrium conditions, and Coulomb's friction laws, and demonstrate their relevance to Civil and Mechanical Engineering applications. | L1 (Remember)<br>L2 (Understand) |
| **CO2** | Apply the principles of force systems, free body diagrams, and equilibrium equations to solve coplanar, concurrent, and spatial force problems including trusses, frames, and friction-based mechanisms using analytical and computational methods. | L3 (Apply) |
| **CO3** | Analyze the geometric properties of engineering sections and bodies by calculating centroids, centre of gravity, and moments of inertia for composite shapes, and evaluate their significance in structural design using software tools. | L4 (Analyze) |
| **CO4** | Evaluate the motion and dynamic response of engineering systems using kinematics and kinetics of particles, applying work-energy and impulse-momentum principles and comparing analytical solutions with AI-based simulation outputs. | L4 (Analyze)<br>L5 (Evaluate) |
| **CO5** | Recall and explain the fundamental concepts of stress, strain, Hooke's law, elastic constants, and their inter-relationships, and identify the mechanical behaviour of engineering materials under axial, thermal, and composite loading conditions. | L1 (Remember)<br>L2 (Understand) |

### CO – PO Articulation Matrix

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

| CO | Bloom's Level | PO1 Eng. Knowledge | PO2 Problem Analysis | PO3 Design/ Dev. | PO4 Investigation | PO5 Modern Tools | PO6 Engineer & Society | PO7 Environ. & Sustain. | PO8 Ethics | PO9 Team Work | PO10 Commun. | PO11 PrjMgmt Finance |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L1, L2 | 3 | 2 | 1 | - | - | - | - | - | - | 2 | - |
| CO2 | L3 | 3 | 3 | 2 | 2 | 2 | - | - | - | 1 | 2 | - |
| CO3 | L4 | 3 | 3 | 2 | 2 | 2 | - | - | - | - | 1 | - |
| CO4 | L4, L5 | 3 | 3 | 2 | 3 | 3 | - | - | - | 1 | 2 | - |
| CO5 | L5, L6 | 2 | 3 | 3 | 2 | 3 | 1 | - | 1 | 2 | 2 | 1 |

## Textbooks

6. Timoshenko S., Young D.H., Rao J.V. and Pati S., Engineering Mechanics, McGraw Hill Education, 5th Edition, 2017.
7. Bhavikatti S.S., A Textbook of Engineering Mechanics, New Age International Publications, 4th Edition, 2018.
8. Timoshenko S. and Gere J., Mechanics of Materials, CBS Publishers, 2006.
9. Strengthofmaterials,R.K.Rajput,S.Chand&Co,NewDelhi
10. Strength of Materials, Basavarajaiah B.S. and Mahadevappa P., Universities Press, 3rd Edition, 2010

## Reference Books

7. Hibbeler R.C., Engineering Mechanics: Statics and Dynamics, Pearson Education, 14th Edition, 2022.
8. Meriam J.L. and Kraige L.G., Engineering Mechanics (Statics & Dynamics), John Wiley, 6th Edition, 2008.
9. Shames I.H., Engineering Mechanics: Statics and Dynamics, PHI, 4th Edition, 2002.
10. Beer F.P. and Johnston E.R., Vector Mechanics for Engineers (Statics & Dynamics), McGraw Hill, 12th Edition, 2019.
11. Bhattacharya B., Introduction to Statics and Dynamics, Oxford University Press, 2nd Edition, 2014.
12. Goldberg D.E., Schaum's Outline of Fluid Mechanics and Hydraulics, McGraw Hill, 4th Edition, 2014.

## Web Resources & NPTEL Links

- NPTEL: Engineering Mechanics – https://nptel.ac.in/courses/112105171
- NPTEL: Rigid Body Dynamics – https://nptel.ac.in/courses/112104120
- PhET Simulations (University of Colorado, Free): https://phet.colorado.edu/en/simulations
- MIT OpenCourseWare – Classical Mechanics: https://ocw.mit.edu/courses/8-01sc-classical-mechanics
- Google Colab for Python: https://colab.research.google.com
- OpenSees documentation: https://opensees.berkeley.edu
- Tracker Video Analysis (Free): https://physlets.org/tracker

| Recommended Free & Open-Source AI/Computational Tools |
|---|
| Python (NumPy, SciPy, Matplotlib, scikit-learn) – Available FREE on Google Colab |
| Scilab 6.x – Free, open-source MATLAB alternative for numerical computation (scilab.org) |
| OpenSees – Free, open-source structural/earthquake engineering simulation framework |
| FreeCAD – Free, open-source 3D CAD with section property computation (freecadweb.org) |
| Tracker – Free video-based physics analysis tool for kinematics experiments (physlets.org/tracker) |
| GeoGebra – Free interactive mathematics and physics visualization (geogebra.org) |
| Algodoo – Free 2D physics simulation environment (algodoo.com) |

**I Year II Sem**

| Course Code | ENGINEERING GRAPHICS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3201T | Common to All Branches | 1 | 0 | 4 | 0 | 3 |

**Prerequisites:**Basic Geometry, Fundamental Python Programming

| **Course objectives:** The objectives of the course are to | |
|---|---|
| 1 | Familiarise with the BIS conventions, geometric construction standards, and engineering scales. |
| 2 | **Develop** visualization skills to interpret and project points, lines, and regular planes into 2D drawing sheets. |
| 3 | **Impart**foundational knowledge on projecting regular 3D solids and reading directional geometric orientations. |
| 4 | Enable the spatial reconstruction of sliced solids and flat unfolded developments of various structural shells. |
| 5 | **Introduce** CAD tools combined with generative AI software to construct multi-view isometric representations from 3D models. |

| **Course Outcomes:** On successful completion of the course, the student will be able to, | |
|---|---|
| 1 | Construct engineering curves and scales using traditional drafting geometry validated by automated algorithmic paths. |
| 2 | Solve multi-planar projection instances of linear points, segments, and standard regular planar silhouettes. |
| 3 | Render projections of standard 3D solids and identify variations using structural boundary algorithms. |
| 4 | Draft complex cross-sectional geometries and mapped surface extensions of sliced geometric forms. |
| 5 | Produce composite isometric layouts alongside generative text-to-CAD system alternatives for dynamic prototyping. |

**Unit 1: Introduction to Engineering Graphics, Curves, and Scales**

**Unit Objectives**

1. To instruct students on basic scaling parameters.
2. To Familiarize with continuous geometric drafting styles.

**Unit Outcome:**

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

**Syllabus 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 Module**: 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**

**Unit Objectives**

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

**Unit Outcomes**

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

**Core Syllabus**: 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 Module**: 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**

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

**Unit Outcomes**

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

**Core Syllabus** :

**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 anotherplane.

**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 automated component sorting inside factory staging bays and conveyor streams.

**AI Integration Module**: 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**

1. To Section basic regular solids accurately and project the actual structural profile of the cut face.
2. To develop unfold3D structures into highly accurate flat-pattern sheet configurations optimized for assembly floor environments.

**Unit Outcomes**

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

**Core Syllabus** :

**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.

- **AI Integration Module**: 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.

**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.

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

**Unit Objectives**

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

**Unit Outcomes**

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

**Core Syllabus** :

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 ofobjects including PCB and Transformations using Auto CAD (*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 Module**: 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.

**Learning Resources**

**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**

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

**Video Links & Online Lectures**

- **Classical Drafting Fundamentals**: NPTEL Engineering Graphical Design Course by IIT Kharagpur
- **Computer Vision and Processing**: OpenCV Official Python Bootcamps and Tutorials
- **Generative Design and Modeling Insights**: MIT 6.S192: Deep Generative Modeling Platforms

| **Engineering Graphics** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | 3 | 2 | 1 | - | - | - | - | - | - | 2 | - | 1 |
| CO2 | 3 | 3 | 2 | - | - | - | - | - | - | 2 | - | 1 |
| CO3 | 3 | 3 | 2 | 1 | - | - | - | - | - | 2 | - | 1 |
| CO4 | 3 | 3 | 3 | 1 | 2 | - | - | - | - | 2 | - | 2 |
| CO5 | 3 | 2 | 2 | - | 3 | - | - | - | 1 | 3 | - | 2 |
| **1** (Low/Slight), **2** (Medium/Moderate), and **3** (High/Substantial) | | | | | | | | | | | | |

I Year II Sem

# BASIC SIMULATION LAB

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3203P | I Year II Sem | 0 | 0 | 0 | 2 | 1 |

**Branch:** Common to Mechanical and Allied Branches (Inferred – not specified in the source.) | Regulation: JNTUA R26 (Inferred – assumed consistent with companion syllabus.)

## Course Objectives

- Master core programming concepts using MATLAB and Simulink tailored for mechanical systems.
- Visualize basic mathematical formulas and physical laws through simple coding steps.
- Develop basic graphical models to understand mechanical movements.
- Gain knowledge about MATLAB Simulink & solve engineering problems.
- Integrate Artificial Intelligence and Machine Learning to optimize mechanical engineering designs.

## List of Experiments

**Core Syllabus**

### Exp 1: Introduction to MATLAB Programming

**Core Syllabus:** Introduction to the MATLAB interface (Command Window, Workspace), basic calculator functions, variables, creating simple vectors/arrays, basic arithmetic operations, and simple 2D line plotting.

**Hands-on Experience:** Write a simple script to calculate the area and volume of various mechanical components (cylinders, spheres) and plot a basic sine wave.

**AI Applications:** Code explanation, fixing basic syntax typos, and generating simple code templates.

**AI Integration:** Use MATLAB AI Chat Playground to ask: "Explain what this basic error message means and how to fix it."

**Video Resources:** MATLAB Onramp (MathWorks – Free 2-Hour Beginner Course).

### Exp 2: Curve Fitting – Straight Line Fit

**Core Syllabus:** Concept of a line of best fit (y = mx + c), basic data importing from Excel, and introduction to the interactive Curve Fitting App.

**Hands-on Experience:** Import data from a standard Hooke's Law experiment (Force vs. Extension) and find the stiffness (k) of a spring using the visual app interface.

**AI Applications:** Spotting outliers or bad data points in an experimental dataset.

**AI Integration:** Use the interactive Curve Fitting App to automatically generate the linear code without writing it from scratch.

**Video Resources:** How to Use the Curve Fitting App (MathWorks YouTube).

### Exp 3: Curve Fitting – Polynomial Fit

**Core Syllabus:** Moving beyond straight lines, understanding non-linear trends, basic concept of a quadratic equation curve, and evaluating fit quality visually.

**Hands-on Experience:** Fit a smooth curve to a car's displacement-time data to visually observe acceleration trends.

**AI Applications:** Simple pattern matching and trend prediction based on historical data.

**AI Integration:** Use the Regression Learner App to explore how a machine learning model can "train" on points to guess a curve.

**Video Resources:** Basic Curve Fitting Techniques (MathWorks).

### Exp 4: Numerical Integration and Differentiation – Trapezoidal Method

**Core Syllabus:** Understanding integration as the simple "area under a curve", dividing an area into basic strips, using the built-in trapz command.

**Hands-on Experience:** Estimate the total work done by a constant or changing force acting over a distance by calculating the area under a Force-Displacement graph.

**AI Applications:** Approximating geometry dimensions from digital sensor data streams.

**AI Integration:** Use generative AI tools to write the logic for a basic for-loop that adds up the areas of separate shapes.

**Video Resources:** Concept of Integration and Trapezoidal Rule (Khan Academy).

### Exp 5: Numerical Integration and Differentiation – Simpson Method

**Core Syllabus:** Introduction to Simpson's rule as a smarter way to find areas using curved tops instead of flat trapezoids, comparing accuracy.

**Hands-on Experience:** Find the cross-sectional area of an irregular mechanical bracket or a river bed profile using standard measurement intervals.

**AI Applications:** Smart shape recognition and automated area estimation in basic design software.

**AI Integration:** Compare results from the Trapezoidal script and Simpson script using an AI assistant to analyze which method is more accurate.

**Video Resources:** Simpson's Rule Made Easy (NPTEL / Online Tutorials).

### Exp 6: Linear and Nonlinear Equations

**Core Syllabus:** Solving a system of basic linear equations (like finding two unknown forces), finding roots of an equation graphically, using simple commands like roots or fzero.

**Hands-on Experience:** Find the precise point where a projectile hits the ground by calculating the zero-crossing root of its trajectory equation.

**AI Applications:** Finding optimal balance points in manufacturing systems or structural designs.

**AI Integration:** Use the interactive Optimization Tool wizard to find the minimum weight of a box without manually calculating derivatives.

**Video Resources:** Solving Equations in MATLAB (MathWorks Tutorial).

### Exp 7: Ordinary Differential Equations & Transforms

**Core Syllabus:** Introduction to differential equations as "rates of change" (e.g., speed changing over time), step-by-step calculation using basic Euler logic, using the ode45 app wizard.

**Hands-on Experience:** Simulate a hot cup of coffee cooling down over time according to Newton's Law of Cooling.

**AI Applications:** Simple time-series trend forecasting (e.g., predicting when a component will overheat).

**AI Integration:** Use the Live Editor Task for Solving ODEs to define and solve the cooling rate visually without deep programming.

**Video Resources:** Introduction to ODEs in MATLAB (MathWorks).

### Exp 8: Engineering Applications Analysis (Basic Mechanics Focus)

**Core Syllabus:** Simulating fundamental concepts from First-Year Engineering Mechanics: static equilibrium, basic lever arms, and simple velocity profiles.

**Hands-on Experience:** Plot how the velocity of a falling object changes over time when affected by gravity and simple air resistance.

**AI Applications:** Identifying unexpected changes or failures in basic mechanical setups.

**AI Integration:** Use the System Identification App to upload data from a basic pendulum swing and let the tool guess its length and drag.

**Video Resources:** Engineering Mechanics Animations using MATLAB (Online Lectures).

### Exp 9: MATLAB Simulink – Introduction to Graphical Modeling

**Core Syllabus:** Introduction to the drag-and-drop Simulink interface, working with basic blocks (Sources, Sinks, Gains, Integrators), and connecting signals visually.

**Hands-on Experience:** Build a visual block diagram model of a basic water tank filling up over time, without writing any lines of code.

**AI Applications:** An introduction to modern digital twins and graphical system simulations.

**AI Integration:** Use Simulink Onramp interactive widgets to discover how pre-trained machine learning blocks can predict the output of a system.

**Video Resources:** Simulink Onramp (MathWorks – Free 2-Hour Visual Guide).

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Develop custom computational scripts to automate data processing and data analysis. | L3 (Apply) |
| CO2 | Plot clear 2D and 3D graphs to analyze and present experimental engineering data. | L3 (Apply) |
| CO3 | Solve basic algebraic and simple differential equations using guided software tools. | L3 (Apply) |
| CO4 | Understand how modern AI assistants can be used ethically to troubleshoot code. | L2 (Understand) |
| CO5 | Apply machine learning toolboxes to predict structural failures and design smart control loops. | L3 (Apply) |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L3 | 3 | 2 | - | - | 3 | - | - | - | - | - | - | 2 |
| CO2 | L3 | 3 | 3 | 2 | 1 | 3 | - | - | - | - | - | - | 1 |
| CO3 | L3 | 3 | 3 | 3 | 2 | 3 | - | - | - | - | - | - | 2 |
| CO4 | L2 | 2 | 3 | 2 | 2 | 3 | - | - | - | - | - | - | 1 |
| CO5 | L3 | - | - | - | - | - | - | - | - | 1 | 3 | - | 2 |

## Textbooks

- Holly Moore, MATLAB for Engineers (6th Edition) — designed for first-year engineering students; covers environment setup, script writing, matrix tracking, arrays, and basic plotting.
- Agam Kumar Tyagi, MATLAB and SIMULINK for Engineers — explains coding logic alongside graphical block layout building.
- Dr. Shailendra Jain, Modeling and Simulation using MATLAB – Simulink — guides beginners through visual modeling, scopes, and signal connections.

## Reference Books

- Steven C. Chapra & Raymond P. Canale, Numerical Methods for Engineers, McGraw Hill Education India.
- John H. Mathews & Kurtis D. Fink, Numerical Methods Using MATLAB.
- Won Y. Yang, Wenwu Cao, Tae-Sang Chung & John Morris, Applied Numerical Methods Using MATLAB.

## Recommended Free & Open-Source Simulation / AI Tools

**Language & Environment:** MATLAB, Simulink – scripting, curve fitting, numerical methods, and graphical modelling.

**AI Assistants:** MATLAB AI Chat Playground – code explanation and debugging support.

**Interactive Apps:** Curve Fitting App, Regression Learner App, Optimization Tool, System Identification App, Live Editor Task for ODEs – guided, low-code problem solving.

**Video Resources:** MATLAB Onramp, Simulink Onramp, NPTEL, Khan Academy – free guided video tutorials.

I Year II Sem

# PHYSICS LAB FOR MECHANICAL ENGINEERS

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0201P | I PHYSICS LAB FOR MECHANICAL ENGINEERS | 0 | 0 | 0 | 3 | 1.5 |

**Branch:** Mechanical Engineering (ME) | Regulation: JNTUA R26 (per source file name; not stated as a labelled field in the source document.)

## Course Objectives

- Determine optical properties of materials (dispersive power, refractive index, wavelength) using prisms, diffraction gratings, and interference techniques.
- Study wave phenomena in vibrating strings, tuning forks, and coupled oscillators.
- Investigate elastic and mechanical properties of materials using Hooke's law and torsional pendulum methods.
- Apply quantum concepts by estimating Planck's constant through the photoelectric effect.
- Employ laser-based and ultrasonic techniques to measure particle size and acoustic velocity in liquids.

## List of Experiments

### 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 using a diffraction grating
- 6. Determination of particle size using a diode laser
- 7. Estimation of Planck's constant using the photoelectric effect
- 8. Determination of spring constant by Hooke's law
- 9. Melde's experiment: Determination of the frequency of the tuning fork
- 10. Sonometer: Verification of the three laws of stretched strings
- 11. Determination of the frequency of normal modes of the Coupled Oscillator
- 12. Study of the Refractive Index of Materials using a hollow prism used in surveying
- 13. Study of oscillations of a mass under different combinations of springs (Loaded Springs)
- 14. Determination of ultrasonic velocity in liquid (Acoustic grating)
- 15. Determination of Rigidity modulus of the material of a wire, dynamic method (Torsional pendulum)

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

## Online Learning Resources

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

## Recommended AI Tools

Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for wave optics and signal analysis

TensorFlow / Keras (free, open-source) – For crack and defect detection CNNs on Google Colab scikit-learn (free, open-source) – For anomaly detection in vibration and NDT data

PhET Interactive Simulations (free) – For wave optics, quantum mechanics and oscillations

**I Year II Sem**

| Course Code | COMMUNICATIVE ENGLISH LAB | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0201P | Common to All Branches | 0 | 0 | 0 | 2 | 1 |

## 1. COURSE OBJECTIVES

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

## 2. COURSE OUTCOMES

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

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

## 3. LAB SYLLABUS – 12 TOPICS WITH AI INTEGRATION

| S.No | Lab Topic | Lab Activities | AI Activity | AI Tools | COs |
|---|---|---|---|---|---|
| 1 | **Phonetics & Pronunciation Practice** | Record & playback vowel/consonant articulation; IPA chart drill; Minimal pair exercises | Use AI speech-to-text to detect mispronunciations; real-time phoneme 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 fallacy identification | AI generates arguments on both sides; evaluate logical structure using ChatGPT | ChatGPT, Claude.ai, Kialo Edu | **CO2, CO3** |
| 11 | **Report Writing & Technical Communication** | Lab report, incident report, and project report formats; headings, data presentation | AI improves report coherence and passive/active voice usage; grammar and structure feedback | Grammarly Business, ChatGPT, Hemingway Editor | **CO4, CO5** |
| 12 | **Etiquette, Netiquette & Personal Branding** | Workplace conversation role play; LinkedIn profile creation; professional social media etiquette | AI reviews LinkedIn summaries; digital persona analysis; personal brand statement via ChatGPT | ChatGPT, Canva AI, LinkedIn AI features | **CO3, CO4, CO5** |

## 4. TEXT BOOKS

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

## 5. REFERENCE BOOKS

| S.No | Reference |
|---|---|
| 1 | Swan, Michael. Practical English Usage (4th Ed.). Oxford University Press, 2016. |
| 2 | Murphy, Raymond. English Grammar in Use (5th Ed.). Cambridge University Press, 2019. |
| 3 | Rizvi, M. Ashraf. Effective Technical Communication (2nd Ed.). McGraw-Hill Education, 2018. |
| 4 | Turton, N.D. & Heaton, J.B. Longman Dictionary of Common Errors. Pearson Longman, 2017. |
| 5 | Lesikar, Raymond V. & Flatley, Marie E. Basic Business Communication (12th Ed.). McGraw-Hill, 2020. |
| 6 | Rutherford, Andrea J. Basic Communication Skills for Technology (3rd Ed.). Pearson, 2019. |

## 6. ONLINE RESOURCES

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

## 7. YOUTUBE CHANNELS

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

## 8. NPTEL / SWAYAM COURSES

| Course Name | Platform | Provider | Link |
|---|---|---|---|

| | | | |
|---|---|---|---|
| Developing Soft Skills and Personality | NPTEL / SWAYAM | IIT Kanpur | https://swayam.gov.in |
| Business English Communication Suite | Coursera (Free Audit) | University of Washington | https://coursera.org |
| Speak English Professionally: In Person, Online & On the Phone | Coursera | Georgia Tech | https://coursera.org |
| English and Academic Preparation | NPTEL | IIT Bombay | https://nptel.ac.in |
| Technical English for Engineers | NPTEL / SWAYAM | IIT Roorkee | https://swayam.gov.in |

## 9. LAB SOFTWARE & AI TOOLS REQUIRED

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

## 10. CO – PO / PSO MAPPING

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

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | - | - | 2 | - | 1 | 2 | - | - | - | - | - | 3 | 2 |
| CO2 | 2 | - | - | 2 | - | 2 | 1 | - | - | - | - | - | 2 | 3 |
| CO3 | 2 | - | - | 3 | - | 3 | 2 | - | - | - | - | - | 2 | 3 |
| CO4 | 2 | - | - | 2 | - | 2 | 1 | - | - | - | - | - | 3 | 2 |
| CO5 | 1 | - | - | 2 | 3 | 2 | 2 | - | - | - | - | - | 2 | 3 |
| Average | 2.0 | - | - | 2.2 | 3.0 | 2.0 | 1.6 | - | - | - | - | - | 2.4 | 2.6 |

## Mapping Legend

| PO | Programme Outcome Description | Correlation with CEL | Strength |
|---|---|---|---|
| **PO1** | Engineering Knowledge / Disciplinary Knowledge | Grammar, vocabulary & technical language | 2 |
| **PO4** | Teamwork & Communication | GD, presentations, professional writing | 3 |
| **PO5** | Modern Tool Usage | AI tools for language learning | 3 |
| **PO6** | Professional Ethics | Workplace etiquette & netiquette | 2 |
| **PO7** | Life-long Learning | Self-paced AI-assisted improvement | 2 |

**I Year II Sem**

| Course Code | HEALTH & WELLNESS, YOGA AND SPORTS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0201L | 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 Outcomes (COs)

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

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

## Unit-wise Syllabus with Learning Outcomes

| UNIT 1 | Health, Wellness and Preventive Healthcare | 5 Hrs |
|---|---|---|
| **Topics** | • Concept of health, wellness and fitness – definitions and seven dimensions<br>• WHO definition of health; holistic health model (Physical, Mental, Social, Spiritual)<br>• Nutrition basics – macronutrients, micronutrients, balanced diet, BMI calculation<br>• Lifestyle diseases – diabetes, hypertension, obesity – causes and prevention<br>• Mental health & stress management – causes, symptoms, COPE model<br>• Sleep hygiene and circadian rhythm regulation<br>• Substance abuse – awareness, prevention and rehabilitation<br>• Personal hygiene, sanitation and communicable disease prevention<br>• Ergonomics and occupational health for students | |
| **Unit-wise Learning Outcomes (ULOs)** | **On completion of this unit, students will be able to:**<br>• ULO1.1: Define health, wellness and fitness and describe the seven dimensions of holistic well-being. [Remember]<br>• ULO1.2: Explain the role of balanced nutrition, BMI and sleep hygiene in | |

| | |
|---|---|
| | maintaining good health. [Understand]<br>• ULO1.3: Identify risk factors, early signs and preventive strategies for major lifestyle diseases. [Understand]<br>• ULO1.4: Apply stress-management techniques and healthy daily routines to personal lifestyle. [Apply]<br>• ULO1.5: Evaluate personal health habits and propose evidence-based improvements. [Evaluate] |

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

| UNIT 3 | Physical Fitness, Sports Skills and Sports Technology | 5 Hrs |
|---|---|---|
| Topics | • Physical fitness components – strength, endurance, flexibility, agility, speed, coordination<br>• Principles of training<br>• – overload, progression, specificity, reversibility, individuality<br>• Warm-up, cool-down and dynamic stretching routines<br>• Indoor sports: Badminton, Table Tennis, Chess, Carrom – rules and basic skills<br>• Outdoor sports: Cricket, Football, Volleyball, Kabaddi, Athletics – rules and skills<br>• Sports injuries – types, RICE method, prevention and return-to-play protocol<br>• Sports psychology – motivation, goal setting, concentration and visualisation<br>• Gender equity and inclusivity in sports; Para-sports awareness<br>• Sports technology – wearables, fitness trackers, GPS analytics, performance dashboards | |
| Unit-wise Learning Outcomes (ULOs) | **On completion of this unit, students will be able to:**<br>• ULO3.1: Identify the components of physical fitness and explain training | |

| | |
|---|---|
| | principles for improving each. [Understand]<br>• ULO3.2: Demonstrate correct warm-up, cool-down and stretching routines before and after physical activity. [Apply]<br>• ULO3.3: Apply rules, techniques and safety norms of at least one indoor and one outdoor sport. [Apply]<br>• ULO3.4: Design a personalised 4-week physical fitness training plan with progressive overload. [Create]<br>• ULO3.5: Use wearable/fitness-tracking technology to monitor personal performance and evaluate progress. [Evaluate] |

### Text Books

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

### Reference Books

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

### Web Resources

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

### YouTube Resources

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

### AI-Integrated Activities

| S.No | AI Activity | Description / Tool |
|---|---|---|
| 1 | AI Fitness Assessment | Log BMI, steps and sleep for 2 weeks using Google Fit / Apple Health; |

| | | |
|---|---|---|
| | Dashboard | 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. |

CO – PO / PSO Mapping

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

| CO / PO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **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 | - |

**I Year II Sem**

| Course Code | CYBERSECURITY AWARENESS FOR ENGINEERS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26ACS5201M | Common to All Branches | 2 | 0 | 0 | 0 | 0 |

COURSE OBJECTIVES

The course aims to:

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

**COURSE OUTCOMES**

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

CO1: Explain fundamental cybersecurity concepts and their importance in digital environments.

CO2: Identify common cyber threats, cyberattacks, and social engineering techniques.

CO3: Apply safe practices for secure use of devices, networks, email, cloud services, and online platforms.

CO4: Demonstrate awareness of cyber ethics, privacy protection, and cyber laws.

CO5: Develop cyber hygiene habits and respond appropriately to common cybersecurity incidents.

**UNIT–I**
**INTRODUCTION TO CYBERSPACE AND CYBERSECURITY**

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.

**UNIT–II**
**CYBER THREATS, CYBERCRIME AND SOCIAL ENGINEERING**

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.

## UNIT–III
## PERSONAL CYBER SAFETY AND CYBER SECURITY TECHNOLOGIES

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.

## UNIT–IV
## PRIVACY, CYBER ETHICS AND CYBER LAWS

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.

## UNIT–V
## CYBER HYGIENE AND SECURE DIGITAL PRACTICES

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.

## TEXT BOOKS

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

## REFERENCE BOOKS

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

## ONLINE RESOURCES

1. National Cyber Security Awareness Portal (India)
2. CERT-In (Indian Computer Emergency Response Team)
3. National Cyber Crime Reporting Portal
4. NIST Cybersecurity Framework Resources
5. OWASP Foundation

Cisco Networking Academy – Introduction to Cybersecurity

**II Year I Sem**

| Course Code | COMPLEX VARIABLES AND PROBABILITY THEORY | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0303T | 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–Thomsonconstruction.

**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 modulustheorem.

**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 theresidue 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | L1 | 3 | 1 | - | - | - | - | - | - | - | - | - |
| **CO2** | L3 | 3 | 2 | 1 | - | - | - | - | - | - | - | - |
| **CO3** | L4 | 3 | 3 | 2 | 2 | 1 | - | - | - | - | - | - |
| **CO4** | L2 | 3 | 2 | - | 1 | - | - | - | - | - | - | - |
| **CO5** | L5 | 3 | 3 | 1 | 2 | 1 | 1 | - | - | 1 | 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.

| PO11 | PO10 | PO9 | PO8 | PO7 | PO6 | PO5 | PO4 | PO3 | PO2 | PO1 | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0.2 | 0.2 | 0 | 0 | 0.2 | 0.4 | 1.0 | 0.8 | 2.2 | 3.0 | Avg. |

## Textbooks

1. E. Kreyszig,*Advanced Engineering Mathematics*, 10th Edition, Wiley, 2011.
2. R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye,*Probability& Statistics for Engineers and Scientists*, 9th Edition, Pearson, 2016.
3. R. V. Churchill and J. W. Brown,*Complex Variables and Applications*, 9th Edition, McGraw-Hill, 2013.
4. J. Miller and J. E. Freund,*Probability and Statistics for Engineers*, 9th Edition, Pearson, 2014.
5. R. K. Jain and S. R. K. Iyengar,*Advanced Engineering Mathematics*, 5th Edition, Alpha Science International, 2021.

## Reference Books

6. S. M. Ross,*Introduction to Probability and Statistics for Engineers and Scientists*, 5th Edition, Academic Press, 2014.
7. B. S. Grewal,*Higher Engineering Mathematics*, 44th Edition, Khanna Publishers, 2021.
8. Dennis G. Zill,*A First Course in Complex Analysis with Applications*, 3rd Edition, Jones & Bartlett, 2018.
9. S. C. Gupta and V. K. Kapoor,*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 |
|---|---|---|---|---|---|---|
| 26AHS0302T | 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.

## Detailed Syllabus

### UNIT 1: Introduction to Value Education and Human Consciousness

**Core Concepts:**

- 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 Concepts:**

- 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 Concepts:**

- 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 Concepts:**

- 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 Concepts:**

- 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

| CO | Course Outcome Statement | Bloom's Level | Bloom's Domain |
|---|---|---|---|
| **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 – Remember / L2 – Understand | Cognitive & Affective |
| **CO2** | Apply the framework of values-based living to personal relationships, family harmony, and societal well-being in daily life contexts. | L3 – Apply | Affective |
| **CO3** | Analyse ethical dilemmas in engineering practice using the lens of universal human values and differentiate value-driven from interest-driven conduct. | L4 – Analyse | Cognitive & Affective |
| **CO4** | Evaluate current engineering and social practices for their alignment with sustainability, ecological balance, and universal human order. | L5 – Evaluate | Cognitive & Affective |
| **CO5** | Create a personal value charter and an action plan for ethical engineering practice using AI reflection tools and collaborative dialogue. | L6 – Create | Affective & Cognitive |

## 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 | - | - | - | - | - | 2 | 2 | 3 | 1 | 2 | - | 2 | 1 | 1 |
| CO2 | - | - | - | - | - | 3 | 2 | 3 | 2 | 2 | 1 | 2 | 1 | 1 |
| CO3 | 1 | 2 | - | 1 | - | 3 | 2 | 3 | 2 | 2 | 1 | 3 | 1 | 2 |
| CO4 | 1 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 2 | 2 |
| CO5 | 1 | 2 | 2 | 1 | 2 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 2 | 2 |
| **Average** | **1.0** | **2.0** | **2.0** | **1.3** | **2.0** | **2.8** | **2.2** | **3.0** | **2.0** | **2.2** | **1.5** | **2.6** | **1.4** | **1.6** |

### Text Books

1. T1. 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. T2. Nagraj, A. (2015). Human Values and Professional Ethics: Universal Human Order. Jeevan Vidya Prakashan, Amarkantak.

### Reference Books

1. R1. Subramanian, C.V. (2020). Professional Ethics and Human Values. Oxford University Press.
2. R2. Bajpai, Sarvesh Chandra. (2019). Engineering Ethics: Concepts and Cases. McGraw-Hill Education.
3. R3. Flood, R.L. (2018). Rethinking the Fifth Discipline: Learning within the Unknowable. Routledge.
4. R4. Covey, Stephen R. (2020). The 7 Habits of Highly Effective People (30th Anniversary ed.). Simon & Schuster.

5. R5. Krishnamurti, J. (2018). Education and the Significance of Life. Harper Collins.
6. R6. Singer, Peter. (2021). Practical Ethics (4th ed.). Cambridge University Press.

### Web Resources

- 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 / Video Resources

- YouTube: NPTEL – Human Values & Professional Ethics – www.youtube.com/c/nptel
- YouTube: TED-Ed – Ethics and Moral Philosophy – www.youtube.com/@TED-Ed
- YouTube: Michael Sandel – Justice: What's the Right Thing to Do? – search on YouTube
- YouTube: AI Ethics Overview – IBM Technology – www.youtube.com/@IBMTechnology
- YouTube: Crash Course – Philosophy – www.youtube.com/@crashcourse

**II Year I Sem**

| Course Code | MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AMB2301T | 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:

Upon completion of this course, students will be able to:

- 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.

## Detailed Syllabus :

### UNIT I: Introduction to Managerial Economics

*Topics: 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.*

**After completing this unit, students will be able to:**

- 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

*Topics: 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).

**After completing this unit, students will be able to:**

- 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 / LibreOfficeCalc (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

*Topics: 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.*

**After completing this unit, students will be able to:**

- 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

*Topics: 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).

**After completing this unit, students will be able to:**

- 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

*Topics: 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.*

**After completing this unit, students will be able to:**

- 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.

## PRESCRIBED TEXT BOOKS

| S.No. | Title | Author(s) | Publisher | Edition / Year |
|---|---|---|---|---|
| 1 | Managerial Economics: Theory and Applications | Mithani D.M | HimalayaPublishing House | 10th Edition, 2025 |
| 2 | Financial Management | Pandey, I.M. | Vikas Publishing House, New Delhi. | 12th Edition, 2022 |
| 3 | Artificial Intelligence: Principles and Applications | Chandramouli Subramanian &Asha A George | Universities Press Private Limited, HYD. | First Edition,2026 |

## REFERENCE BOOKS

| S.No. | Title | Author(s) | Publisher | Edition / Year |
|---|---|---|---|---|
| 1 | Managerial Economics and Financial Analysis | Aryasri A.R. | McGraw-Hill Education | Revised Edition, 2020 |
| 2 | Financial Management: Theory and Practice | Prasanna Chandra | Tata McGraw-Hill, New Delhi | 10th Edition, 2019 |
| 3 | Business Analytics: The Science of Data-Driven Decision Making | Kumar U.D. | Wiley India | 2nd Edition, 2021 |
| 4 | Engineering Economics and Financial Accounting | Siddiqui, S. A. | S. Chand Publishing, New Delhi | 3rd Edition, 2019 |
| 5 | Economics for Engineers | Gupta, G. S. | Tata McGraw-Hill Education | 2nd Edition, 2017 |

## Online Resources:

| S.No. | Platform | Course / Resource | Link / Access |
|---|---|---|---|
| 1 | NPTEL (IIT) | Managerial Economics | https://nptel.ac.in/courses/110105063 |
| 2 | NPTEL (IIT) | Financial Accounting for Engineers | https://nptel.ac.in/courses/110107114 |
| 3 | Coursera | Economics for Non-Economists (Univ. of California) | https://www.coursera.org/learn/economics |
| 4 | edX | Financial Accounting (MIT OpenCourseWare) | https://www.edx.org/course/financial-accounting |
| 5 | Khan Academy | Microeconomics & Finance Basics | https://www.khanacademy.org/economics-finance-domain |
| 6 | SWAYAM Portal | Business Economics (UGC-approved) | https://swayam.gov.in |

# AI TOOLS USED IN THIS COURSE

| S.No. | AI Tool | Application in Course |
|---|---|---|
| 1 | ChatGPT / Claude AI | Concept clarification, case study generation, problem solving assistance |
| 2 | Microsoft Copilot | Financial analysis, spreadsheet automation, data interpretation |
| 3 | Google Gemini | Research support, market structure comparisons, summarisation |
| 4 | Tally AI / Zoho Books AI | Digital accounting simulations, ledger preparation, journal entries |
| 5 | Statista / Tableau AI | Data visualisation for demand trends and financial ratios |
| 6 | Wolfram Alpha | Quantitative problems – NPV, IRR, Break-Even, elasticity calculations |
| 7 | Grammarly AI | Report writing, case study documentation, assignments |

# AI-INTEGRATED ACTIVITIES

| 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

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

## 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 | 2 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 1 | 1 |
| CO2 | 2 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 2 | 3 | 2 | 2 | 2 |
| CO3 | 2 | 3 | 2 | 3 | 3 | 2 | 1 | 1 | 2 | 2 | 3 | 3 | 2 | 2 |
| CO4 | 2 | 3 | 2 | 3 | 3 | 2 | 1 | 1 | 2 | 2 | 3 | 3 | 2 | 2 |
| CO5 | 2 | 3 | 3 | 3 | 3 | 2 | 2 | 1 | 3 | 3 | 3 | 3 | 3 | 3 |
| **Average** | **1.8** | **2.8** | **2.0** | **2.4** | **2.4** | **2.0** | **1.2** | **1.0** | **1.8** | **2.2** | **2.8** | **2.6** | **2.0** | **2.0** |

## 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.

| Course Code | BASIC ELECTRICAL & ELECTRONICSENGINEERING | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AEC304T | Common to Civil &Mechanical Branches | 3 | 0 | 0 | 0 | 3 |

**II Year I Sem**

II Year –I Sem

## PARTA:BASICELECTRICALENGINEERING

Course Objectives:

1. To expose students to the fundamentals of electrical engineering, electrical laws, and circuit analysis techniques.
2. To impart knowledge on the construction, operation, and applications of electrical machines and measuring instruments.
3. To familiarize students with conventional and non-conventional energy resources, electricity billing concepts, and electrical safety measures.
4. To develop analytical and problem-solving skills related to AC/DC circuits, electrical machines, and power systems.
5. To introduce AI-assisted tools and intelligent techniques for electrical system analysis, monitoring, simulation, and energy management.

Course Outcomes:

| CO No. | Course Outcome | BL |
|---|---|---|
| CO1 | Remember the fundamental laws, electrical quantities, operating principles of electrical machines, measuring instruments, and energy systems. | L1 |
| CO2 | Understand AC/DC circuit behavior, operation of machines and measuring instruments, electricity billing concepts, power generation methods, and electrical safety practices. | L2 |
| CO3 | Apply mathematical methods, electrical engineering principles, and AI-assisted tools to solve problems related to circuits, machines, and energy systems. | L3 |
| CO4 | Analyze electrical circuits, machine performance, measuring instruments, and power systems using conventional and AI-based techniques. | L4 |
| CO5 | Analyze electrical system performance, energy utilization, protection methods, and intelligent monitoring systems for efficient and safe operation. | L4 |

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

Note: 1-Low, 2-Medium, 3-High

**Syllabus**

**UNITI DC&ACCircuits**

**DC Circuits:** Electrical circuit elements (R, L and C), Ohm's Law and its limitations, KCL & KVL, series, parallel, series-parallel circuits, Super Position theorem, Simple Numerical problems.

**AC Circuits:** A.C. Fundamentals: Equation of AC Voltage and current, waveform, time period, frequency, amplitude, phase, phase difference, average value, RMS value, form factor, peak factor, Voltageand current relationship with phasordiagrams in R, L, and C circuits, Concept of Impedance, Analysis of R-L, R-C, R-L-C Series circuits, Active power, reactive power and apparent power, Concept of power factor (Simple Numerical problems).

**AI-Integrated Topics**

- Introduction to AI-assisted Electrical Circuit Analysis
- Intelligent Waveform and Phasor Diagram Analysis
- Machine Learning Applications for Voltage and Current Prediction
- AI-based Power Factor Monitoring and Correction
- AI-enabled Circuit Simulation Tools and Applications

**Suggested AI Tools**

- MATLAB
- PSpice
- LTspice
- Python

**UNITIIMachinesandMeasuring Instruments**

**Machines:** Construction, principle and operation of (i) DC Motor, (ii) DC Generator, (iii) Single Phase Transformer, (iv) Three Phase Induction Motor and (v) Alternator, Applications of electrical machines.

**MeasuringInstruments:**ConstructionandworkingprincipleofPermanentMagnetMovingCoil (PMMC), Moving Iron (MI) Instruments and Wheat Stone bridge.

**AI-Integrated Topics**

- Introduction to AI in Electrical Machines
- AI-based Predictive Maintenance of Motors and Transformers
- Smart Sensor-based Monitoring of Electrical Parameters
- AI-based Error Detection in Measuring Instruments
- Intelligent Digital Measuring Instruments

**Suggested AI Tools**

- MATLAB Simulink
- TensorFlow
- LabVIEW
- Scikit-learn

**UNITIIIEnergyResources,ElectricityBill&Safety Measures**

**Energy Resources:** Conventional and non-conventional energy resources; Layout and operation of various Power Generation systems: Hydel, Nuclear, Solar & Wind power generation.

**Electricity Bill:** Power rating of household appliances including air conditioners, PCs, Laptops, Printers, etc. Definition of "unit" used for consumption of electrical energy, two-part electricitytariff, calculation of electricity bill for domestic consumers.

**Equipment Safety Measures:** Working principle of Fuse and Miniature Circuit Breaker (MCB), merits and demerits. Personal safety measures: Electric Shock, Earthing and its types, Safety Precautions to avoid shock.

**AI-Integrated Topics**

- Introduction to Smart Energy Systems and AI Applications
- Smart Metering and Intelligent Electricity Consumption Monitoring
- AI-assisted Electricity Bill Prediction and Energy Analytics
- Intelligent Energy Management Systems for Smart Homes
- Smart Electrical Safety Monitoring Systems

**Suggested AI Tools**

- Power BI
- HOMER Pro
- Arduino IDE
- Google Colab

LearningResources:

**Textbooks:**

1. BasicElectricalEngineering,D.C.Kulshreshtha, TataMcGrawHill,2019, FirstEdition
2. PowerSystemEngineering,P.V.Gupta,M.L.Soni,U.S.BhatnagarandA.Chakrabarti, Dhanpat Rai & Co, 2013
3. FundamentalsofElectricalEngineering,RajendraPrasad,PHIpublishers,2014,ThirdEdition

ReferenceBooks:

1. BasicElectricalEngineering,D.P.Kothariand I.J.Nagrath,McGrawHill,2019,Fourth Edition
2. PrinciplesofPowerSystems,V.K.Mehtha,S.ChandTechnicalPublishers,2020
3. BasicElectricalEngineering,T.K.NagsarkarandM.S.Sukhija,OxfordUniversityPress, 2017
4. Basic Electrical and Electronics Engineering, S. K. Bhatacharya, Person Publications, 2018,Second Edition.

WebResources:

1. https://nptel.ac.in/courses/108105053
2. https://nptel.ac.in/courses/108108076

## PART-B:BASIC ELECTRONICS ENGINEERING

**Course Objectives:**

1. Understand semiconductor devices like PN junction diode, Zener diode, BJT and analyse their characteristics and applications in basic electronic circuits.
2. Apply semiconductor device knowledge to design and analyse rectifier circuits, amplifiers, and instrumentation systems.
3. Understand digital electronics fundamentals like number systems, logic gates, Boolean algebra, and sequential circuits.

### UNIT I Semiconductor Devices

Introduction — Evolution of electronics: vacuum tubes to nano electronics. Characteristics of PN Junction Diode. Zener Effect — Zener diode and its characteristics. Bipolar Junction Transistor — CB, CE, CC configurations and characteristics. Elementary treatment of small signal CE amplifier.

**Suggested Tasks**

1. Use AI/LLM tools to explain the construction of a PN junction diode, and the Zener effect; ask it to describe the V-I characteristics of a PN junction and Zener diode with sketches, and compare their operating regions in a short table.
2. Use a provided Python template in Google Colab to plot the ideal V-I characteristic curve of a PN junction diode using the diode equation; vary the saturation current and thermal voltage and observe how the curve shape changes, then superimpose the Zener diode characteristic to compare forward and reverse breakdown behaviour.
3. UseAI/LLM tool to explain CB, CE, and CC BJT configurations — input/output characteristics, current gain, and typical applications; prepare a comparison table covering input impedance, output impedance, current gain, voltage gain, and phase relationship, and ask the LLM to describe the small signal CE amplifier operation in simple terms.

### UNIT II Basic Electronic Circuits and Instrumentation

Block diagram of a DC power supply, working of a full wave bridge rectifier, capacitor filter (no analysis), working of simple Zener voltage regulator. Amplifiers: block diagram of Public Address system, circuit diagram and working of common emitter (RC coupled) amplifier with frequency response, concept of voltage divider biasing. Electronic Instrumentation: block diagram of an electronic instrumentation system.

**Suggested Tasks**

1. Use AI/LLM toolto explain the block diagram of a DC power supply and the working of a full wave bridge rectifier; ask it to describe the role of a capacitor filter and a Zener voltage regulator in simple terms, and prepare a labelled block diagram description with the function of each stage (transformer, rectifier, filter, regulator).
2. Use a provided Python template in Google Colab to simulate a full wave bridge rectifier output: generate a sinusoidal input signal, apply ideal diode rectification, and plot the input and rectified output waveforms; then add a smoothing capacitor and observe how the ripple voltage changes with different capacitor values.
3. Use AI/LLM toolto explain the working of a common emitter RC coupled amplifier — the role of coupling capacitors, bypass capacitor, and voltage divider bias network; ask it to describe the frequency response (low, mid, and high frequency regions) and sketch the gain-vs-frequency curve, then prepare a short note on the block diagram of an electronic instrumentation system.

### UNIT III Digital Electronics

Overview of number systems. Logic gates including universal gates. BCD codes, Excess-3 code, Gray code, Hamming code. Boolean algebra — basic theorems and properties. Truth tables and functionality of logic gates: NOT, OR, AND, NOR, NAND, XOR, and XNOR. Simple combinational circuits: half adder and full adder. Introduction to sequential circuits — flip-flops, registers, and counters (elementary treatment only).

**Suggested Tasks**

1. Use an AI/LLM toolto explain the binary, octal, and hexadecimal number systems and conversions between them; ask it to describe BCD, Excess-3, Gray, and Hamming codes with examples, and solve 2–3 numerical problems on code conversion (e.g., decimal to BCD, binary to Gray code); verify the results manually.
2. Use a provided Python template in Google Colab to generate truth tables for all seven logic gates (NOT, OR, AND, NOR, NAND, XOR, XNOR) for 2-variable and 3-variable inputs; verify De Morgan's theorems programmatically by comparing both sides of the identities, and implement a half adder and full adder using basic gate functions.
3. Use an AI/LLM toolto explain SR, D, JK, and T flip-flops in simple terms — truth tables, excitation tables, and the concept of clock-triggered operation; ask it to describe the working of a 4-bit shift register and a 3-bit ripple counter with state sequences, and prepare a concise comparison table of flip-flop types.

**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**

***At the end of the course the student will be able to***

1. Analyse the characteristics of PN junction diode, Zener diode, and BJT configurations to evaluate their behaviour and select appropriate devices for basic electronic applications.
2. Apply knowledge of rectifiers, filters, amplifiers, and instrumentation systems to explain circuit operation and simulate waveform behaviour using Python tools.
3. Design basic combinational and sequential digital circuits using logic gates, Boolean algebra, and flip-flops, and verify logic behaviour using LLM and Python simulation tools.

**Textbooks**

1. Boylestad, R.L. and Nashelsky, L., "Electronic Devices and Circuit Theory", 11th Edition, Pearson Education, 2013.
2. Floyd, T.L., "Digital Fundamentals", 11th Edition, Pearson Education, 2015.

**Reference Books**

1. Sedra, A.S. and Smith, K.C., "Microelectronic Circuits", 7th Edition, Oxford University Press, 2014.
2. Millman, J. and Halkias, C.C., "Electronics Devices and Circuits", Tata McGraw-Hill, 2008.
3. Morris Mano, M. and Ciletti, M.D., "Digital Design", 5th Edition, Pearson Education, 2013.
4. Razavi, B., "Fundamentals of Microelectronics", 2nd Edition, Wiley, 2013.

# MATERIAL SCIENCE AND METALLURGY

| Course Code | Year/Sem | L T Pr P Credits |
|---|---|---|
| 26AME3301T | B.Tech ME – II Year II Sem | 3 0 0 0 3 |

**Prerequisites:** Engineering Chemistry, Engineering Physics

## Course Objectives

- Demonstrate the crystalline structure of different metals and study the stability of phases in different alloy systems.
- Teach the behaviour of ferrous and non-ferrous metals and alloys and their application in different domains.
- Study the effect of heat treatment and addition of alloying elements on properties of ferrous metals.
- Teach the methods of making metal powders and applications of powder metallurgy.
- Identify the properties and applications of ceramics, composites, and other advanced materials.

## Course Content

### UNIT I: Structure of Metals, Crystallization, and CV Grain Segmentation

**Core Syllabus:** Structure of metals and constitution of alloys: crystallization of metals; packing factor (SC, BCC, FCC, HCP), line density, plane density; grain and grain boundaries, effect of grain boundaries, determination of grain size; imperfections, slip and twinning; necessity of alloying, types of solid solutions, Hume-Rothery's rules, intermediate alloy phases, and electron compounds. Equilibrium diagrams: experimental methods of construction, isomorphous alloy systems, equilibrium cooling and heating of alloys, lever rule, coring, miscibility gaps, eutectic systems, congruent melting intermediate phases, peritectic reaction; transformations in the solid state (allotropy, eutectoid, peritectoid reactions), phase rule, relationship between equilibrium diagrams and properties of alloys; study of binary phase diagrams such as Cu-Ni and Fe-Fe3C.

**Hands-on Practices:**

- Project: a Python script using OpenCV (adaptive thresholding and watershed segmentation) to process an optical micrograph of etched mild steel, isolate individual grains, calculate their surface areas, and output the ASTM grain size
- Application: real-time quality verification of microstructural refinement loops inside industrial rolling mills

**AI Integration:** Computer-vision pipelines for automated metallographic grain-boundary detection, using deep semantic-segmentation networks such as U-Net to segment grains and compute average ASTM grain-size numbers directly from optical microscope images.

**Free/Open-Source AI Tools:** Python with OpenCV (free, open-source) for grain segmentation and ASTM grain-size calculation on Google Colab; open U-Net implementations (PyTorch/TensorFlow, open-source) for semantic segmentation

### UNIT II: Ferrous, Non-Ferrous Metals and Alloys

**Core Syllabus:** Ferrous metals and alloys: structure and properties of white cast iron, malleable cast iron, grey cast iron, spheroidal graphite cast iron, alloy cast iron; classification of steels, structure and properties of plain carbon steels, low alloy steels, Hadfield manganese steels, tool and die steels. Non-ferrous metals and alloys: structure and properties of copper and its alloys, aluminium and its alloys, titanium and its alloys, magnesium and its alloys, super alloys.

**Hands-on Practices:**

- Project: a lightweight CNN classification script (TensorFlow/Keras or PyTorch) trained on cropped micrograph images to classify samples into Gray Cast Iron, Nodular Cast Iron, or Low-Carbon Steel

- Application: automated metallurgical scanning for scrap-metal sorting lines and component failure forensics

**AI Integration:** Deep convolutional neural networks (CNNs) for microstructural classification, training image-classification networks to read microscope scans and instantly label phase balances (e.g. separating pearlite from martensite).

**Free/Open-Source AI Tools:** Python with TensorFlow/Keras or PyTorch (free, open-source) on Google Colab for CNN-based micrograph classification

## UNIT III: Heat Treatment of Steels

**Core Syllabus:** Effect of alloying elements on the Fe-Fe3C system; annealing, normalizing, hardening; TTT diagrams; tempering; hardenability; surface-hardening methods; age-hardening treatment; cryogenic treatment.

**Hands-on Practices:**

- Project: a custom loss function embedding the Johnson-Mehl-Avrami-Kolmogorov (JMAK) kinetics equation, training a neural network to predict the volumetric fraction of transformed austenite during continuous cooling while penalising predictions that break phase-transformation physics
- Application: smart digital-twin controllers for continuous conveyor tempering ovens and industrial induction-hardening tracks

**AI Integration:** Physics-informed neural networks (PINNs) for modelling phase transformations, embedding diffusion-physics laws (e.g. Avrami equation constants) directly into the network's loss function to predict continuous-cooling hardness outcomes.

**Free/Open-Source AI Tools:** Python with PyTorch/TensorFlow (free, open-source) on Google Colab for physics-informed neural network training

## UNIT IV: Powder Metallurgy

**Core Syllabus:** Basic processes; methods of producing metal powders (milling, atomization, granulation, reduction, electrolytic deposition); compacting methods; sintering; methods of manufacturing sintered parts; secondary operations; applications of powder metallurgical products.

**Hands-on Practices:**

- Powder characterization by sieve analysis for particle size distribution
- Flowability testing using a Hall flowmeter
- Apparent versus tap density measurement
- Cold die-compacting of copper/iron powders using hydraulic presses
- Sintering-furnace simulation to observe shrinkage and bonding
- Real-world applications: self-lubricating bearings and high-strength gears (automotive); tungsten-carbide cutting inserts; heat-resistant turbine disks and porous filters (aerospace); titanium porous implants (biomedical); copper-tungsten circuit-breaker contacts (electrical)

**AI Integration:** Generative design that optimises die geometries to reduce structural stress points. Predictive-sintering ML models that forecast final component shrinkage rates. Computer vision that flags micro-cracks in green compacts, neural networks that predict optimal binder-to-powder ratios, and reinforcement learning that adjusts furnace temperature zones in real time.

**Free/Open-Source AI Tools:** Python with scikit-learn (free, open-source) for shrinkage/property prediction models on Google Colab; OpenCV (free, open-source) for green-compact defect detection

## UNIT V: Non-Ferrous Alloys & Advanced Materials

**Core Syllabus:** Copper and its alloys: brasses and bronzes. Aluminium and its alloys: Al-Cu, Al-Si systems. Titanium and its alloys. Composite materials: classification, particle-reinforced, fiber-reinforced, structural composites; applications of composites and smart materials; basics of nano-materials.

**Hands-on Practices:**

- Composite fabrication by vacuum-bagging lay-up of a carbon-fibre PMC
- Micro-hardness testing across cermets and ceramics
- Fracture analysis comparing brittle versus ductile failure under digital microscopes
- Shape-recovery testing of Nitinol wire under heat
- Viscosity measurement of glass-transition behaviour at varying high temperatures
- Real-world applications: carbon-carbon composite heat shields (aerospace); boron-carbide ballistic ceramics (military armor); carbon-ceramic matrix composite brakes (high-performance automotive); alumina-TiC cermet cutting tools; piezoelectric smart ceramics for ultrasound sensors

**AI Integration:** Generative AI for predicting stable atomic structures of new cermets (material discovery). Computer vision for detecting micro-voids in fiber-reinforced layups, ML models forecasting tensile strength from fiber orientation, reinforcement learning tuning actuators from smart-material sensor inputs, and AI algorithms predicting chemical-vapour-deposition yields for carbon nanotubes.

**Free/Open-Source AI Tools:** Python with scikit-learn (free, open-source) for tensile-strength/property prediction on Google Colab; OpenCV (free, open-source) for defect scanning in composite layups; Materials Project API (free) for materials data lookup

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level* |
|---|---|---|
| CO1 | Understand the crystalline structure of different metals and study the stability of phases in different alloy systems. | L1 (Remember) / L2 (Understand) |
| CO2 | Study the behaviour of ferrous and non-ferrous metals and alloys and their application in different domains. | L2 (Understand) |
| CO3 | Understand the effect of heat treatment and addition of alloying elements on properties of ferrous metals. | L2 (Understand) |
| CO4 | Grasp the methods of making metal powders and applications of powder metallurgy. | L2 (Understand) |
| CO5 | Comprehend the properties and applications of ceramics, composites, and other advanced materials. | L2 (Understand) |

**Bloom's levels were not specified in the source syllabus and have been inferred here from the verbs used in each outcome statement; please verify/adjust as needed.*

## CO – PO Articulation Matrix

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

*Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. PSO columns were not present in the source CO-PO matrix and have been omitted; add them here if the department defines PSOs for this programme.*

## Textbooks

1. S.H. Avner, Introduction to Physical Metallurgy, 2nd Edition, Tata McGraw-Hill, 1997.

2. Donald R. Askeland, Essentials of Materials Science and Engineering, 4th Edition, CL Engineering Publications, 2018.

## Reference Books

1. Dr. V.D. Kodgire, Material Science and Metallurgy, 39th Edition, Everest Publishing House, 2017.

2. V. Raghavan, Material Science and Engineering, 5th Edition, Prentice Hall of India, 2004.

3. William D. Callister Jr, Materials Science and Engineering: An Introduction, 8th Edition, John Wiley and Sons, 2009.

4. George E. Dieter, Mechanical Metallurgy, 3rd Edition, McGraw-Hill, 2013.

5. Yip-Wah Chung, Introduction to Material Science and Engineering, 2nd Edition, CRC Press, 2022.
6. A V K Suryanarayana, Material Science and Metallurgy, B S Publications, 2014.
7. U. C. Jindal, Material Science and Metallurgy, 1st Edition, Pearson Publications, 2011.

## Web Resources & Video Links

- NPTEL: https://archive.nptel.ac.in/courses/113/106/113106032/
- edX – Mechanical Behavior of Materials Part 3 (MIT): https://www.edx.org/learn/mechanics/massachusetts-institute-of-technology-mechanical-behavior-of-materials-part-3-time-dependent-behavior
- YouTube lecture: https://www.youtube.com/watch?v=9Sf278j1GTU
- Coursera – Fundamentals of Materials Science: https://www.coursera.org/learn/fundamentals-of-materials-science
- Coursera – Material Behavior: https://www.coursera.org/learn/material-behavior
- NPTEL – Introduction to Materials Science and Engineering, IIT Delhi
- Materials Informatics and Machine Learning – Steve Brunton Lecture Series
- Scikit-Learn Regression Models and Materials Project API Integration Guides

## Recommended Free & Open-Source AI/Computational Tools

- Python with OpenCV (free, open-source) – grain segmentation and defect detection on Google Colab
- TensorFlow/Keras or PyTorch (free, open-source) – CNN and physics-informed neural network modelling
- scikit-learn (free, open-source) – property and shrinkage prediction models
- Materials Project API (free) – materials property database lookup
- Google Colab (free) – hosted environment for all of the above

# MECHANICS OF SOLIDS

| Course Code | Year/Sem | L T Pr P Credits |
|---|---|---|
| 26AME3302T | II Year I Sem | 2 0 2 0 3 |

**Prerequisites:** Engineering Mechanics, Linear Algebra & Calculus

## Course Objectives

- Understand stress-strain relationships in materials.
- Analyze structural members under various loading conditions.
- Analyse beams, columns, and frames for normal, shear, and torsional stresses and solve deflection problems.
- Understand the loads, stresses, and strains acting on a structure and their relations in elastic behaviour.
- Design and analyse industrial components such as pressure vessels.

## Course Content

### UNIT I: Simple Stresses & Strains

**Core Syllabus:** Elasticity and plasticity; types of stresses and strains; Hooke's law; stress-strain diagram for mild steel; working stress; factor of safety; lateral strain, Poisson's ratio and volumetric strain; bars of varying section; composite bars; temperature stresses; complex stresses; stresses on an inclined plane under different uniaxial and biaxial stress conditions; principal planes and principal stresses; Mohr's circle; relation between elastic constants; strain energy, resilience; gradual, sudden, impact, and shock loadings.

**Hands-on Practices:**

- Universal Testing Machine (UTM) experiments
- Stress-strain curve generation for different materials
- Measurement of Poisson's ratio using strain gauges
- Thermal expansion experiments

**AI Integration:** Automated UTM data-acquisition pipeline feeding into Python/MATLAB-based stress-strain curve analysis. ML-based material classifier that identifies material type from its stress-strain curve and predicts likely failure points. Automated test-report generation for material testing, and AI-vision strain measurement using Digital Image Correlation (DIC). A virtual material-testing laboratory allows students to practise specimen tests safely.

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, Matplotlib) on Google Colab for stress-strain analysis; scikit-learn (free, open-source) for material property/failure prediction; OpenCV (free, open-source) for DIC-based strain vision

### UNIT II: Shear Force and Bending Moment

**Core Syllabus:** Definition of beam; types of beams; concept of shear force and bending moment; S.F. and B.M. diagrams for cantilever, simply supported, and overhanging beams subjected to point loads, u.d.l., uniformly varying loads, and combinations of these loads; point of contra-flexure; relation between S.F., B.M., and rate of loading at a section of a beam.

**Hands-on Practices:**

- Physical beam loading experiments
- Strain gauge measurements on loaded beams
- Model beam construction and testing
- Manual SFD/BMD construction practice

**AI Integration:** Python-based automated SFD/BMD generator (e.g. the BeamBending library) for any beam configuration, description, or loading combination. AI drawing-recognition tool that converts hand-sketched beam diagrams into digital SFD/BMD plots for self-verification. Interactive real-time simulator for visualising shear force and bending moment as loads change, and an AI chatbot offering step-by-step beam-analysis guidance.

**Free/Open-Source AI Tools:** Python (SymPy, BeamBending library) on Google Colab; free online structural-analysis/beam calculators for verification

## UNIT III: Bending Stresses and Shear Stresses in Beams

**Core Syllabus:** Flexural stresses: theory of simple bending, derivation of the bending equation, determination of bending stresses, section modulus of rectangular, circular, I, and T sections, design of simple beam sections. Shear stresses: derivation of the formula; shear stress distribution across rectangular, circular, triangular, I, and T sections.

**Hands-on Practices:**

- Photoelastic stress analysis
- Strain gauge measurement of bending stresses
- Section property calculations for standard sections
- Physical beam failure testing

**AI Integration:** ML-based model for optimal beam-section selection given load and material constraints. Python-based custom stress calculator for bending and shear stress distribution across standard sections. FEA-driven stress visualisation (ANSYS/Abaqus) linked to CAD for automatic section-property calculation, generative-AI assisted design of custom beam cross-sections, and a virtual photoelastic analyser for visualising stress patterns.

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, Matplotlib) on Google Colab for section-property and stress calculators; FreeCAD (free) for cross-section visualization in place of proprietary CAD/FEA packages

## UNIT IV: Deflection of Beams and Torsion

**Core Syllabus:** Deflection of beams: bending into a circular arc; slope, deflection, and radius of curvature; differential equation for the elastic line of a beam; double integration and Macaulay's methods; determination of slope and deflection for cantilever and simply supported beams subjected to point loads, UDL, and UVL; Mohr's theorem and the moment-area method applied to simple cases. Torsion: introduction, derivation, torsion of circular shafts, pure shear, transmission of power by circular shafts, shafts in series, shafts in parallel.

**Hands-on Practices:**

- Dial gauge deflection measurements
- Comparison of theoretical and experimental deflections
- Spring constant determination for beams
- Deflection under moving loads

**AI Integration:** Python/SymPy-based automated deflection calculator that performs double integration or Macaulay's method symbolically and recommends the most suitable method for a given beam. AI curve-fitting of experimental dial-gauge data against theoretical deflection curves, a virtual deflection-measurement system, and a predictive ML model for beam deflection under varying load configurations.

**Free/Open-Source AI Tools:** Python (SymPy, NumPy, SciPy) on Google Colab for symbolic/numerical deflection calculations; open-source curve-fitting libraries for experimental data analysis

## UNIT V: Thin Cylinders, Thick Cylinders & Columns

**Core Syllabus:** Thin and thick cylinders: thin seamless cylindrical shells; derivation of formulae for longitudinal and circumferential stresses; hoop, longitudinal, and volumetric strains; changes in diameter and volume of thin cylinders; thin spherical shells; wire-wound thin cylinders; Lame's equation; cylinders subjected to inside and outside pressures; compound cylinders. Columns: buckling and stability, columns with pinned ends, columns with other support conditions, limitations of Euler's formula, Rankine's formula.

**Hands-on Practices:**

- Calculating hoop stress and longitudinal stress in thin cylinders using real-life examples such as pipes or gas cylinders

- Applying Lame's equation to find radial and hoop stress distribution in thick cylinders subjected to internal and external pressures
- Verifying Euler's buckling load for different end conditions such as pinned-pinned and fixed-free columns

**AI Integration:** AI tutors such as ChatGPT for step-by-step derivation of hoop stress, Lame's equations, and Euler's/Rankine's formulas, and for solving numerical problems with explanations. Quick AI-assisted verification of manually computed stresses, strains, and buckling loads. MATLAB/Python-based plotting of stress distribution in thick cylinders and simulation of column buckling under varying end conditions, AI simulation/image-generation tools to visualise stress distribution and buckling shapes, and AI-supported plotting assistants (Python/Desmos AI) for load-versus-deflection curves.

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, Matplotlib) on Google Colab for stress and buckling plots; AI tutors such as ChatGPT/Claude (free tier) for derivations and problem solving; Desmos (free) for load-deflection graphing

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level* |
|---|---|---|
| CO1 | Calculate stress and strain in structural members under axial, torsional, and bending loads. | L3 (Apply) |
| CO2 | Analyze deflection in beams using various methods. | L4 (Analyze) |
| CO3 | Apply the concepts of stress and strain to analyze and design structural members and machine parts under axial, shear, and bending loads and moments. | L3 (Apply) / L6 (Create) |
| CO4 | Understand material behaviour through stress-strain relationships. | L2 (Understand) |
| CO5 | Solve complex design problems such as pressure vessels. | L5 (Evaluate) / L6 (Create) |

**Bloom's levels were not specified in the source syllabus and have been inferred here from the verbs used in each outcome statement; please verify/adjust as needed.*

## CO – PO Articulation Matrix

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

*Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. PSO columns were not present in the source CO-PO matrix and have been omitted; add them here if the department defines PSOs for this programme.*

## Textbooks

1. B.C. Punmia, Strength of Materials, 10th Edition, Laxmi Publications Pvt. Ltd, New Delhi, 2018.
2. G.H. Ryder, Strength of Materials, Palgrave Macmillan Publishers India Ltd, 1961.

## Reference Books

1. Gere and Timoshenko, Mechanics of Materials, 2nd Edition, CBS Publications, 2004.
2. U.C. Jindal, Strength of Materials, 2nd Edition, Pearson Education, 2017.
3. Timoshenko, Strength of Materials Part I & II, 3rd Edition, CBS Publishers, 2004.
4. Andrew Pytel and Ferdinand L. Singer, Strength of Materials, 4th Edition, Longman Publications, 1990.
5. Popov, Mechanics of Solids, 2nd Edition, Pearson Education, 2015.

## Web Resources & Video Links

- NPTEL – Strength of Materials: https://nptel.ac.in/courses/105/105/105105108/
- NPTEL Online Courses: https://onlinecourses.nptel.ac.in

- Poriyaan – Strength of Materials notes: https://www.poriyaan.in/paper/strength-of-materials-94/

## Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, Matplotlib, SymPy) – free on Google Colab for stress, deflection, and buckling calculations
- scikit-learn (free, open-source) – material property and failure prediction
- OpenCV (free, open-source) – Digital Image Correlation (DIC) strain vision
- FreeCAD (free, open-source) – cross-section visualization
- AI tutors such as ChatGPT/Claude (free tier) – derivations, concept clarification, and problem solving
- Desmos (free) – load-deflection and stress-distribution graphing

II Year I Sem

| Course Code | BASIC ELECTRICAL & ELECTRONICSENGINEERINGLAB | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AEC304P | Common to Civil &Mechanical Branches | 0 | 0 | 0 | 2 | 1 |

## PART-A: ELECTRICAL ENGINEERING

**Course Objectives:**

- To provide practical knowledge of fundamental electrical laws, network theorems, measuring instruments, and electrical machines through laboratory experiments.
- To develop skills in performing electrical measurements, analyzing circuit behavior and evaluating machine characteristics using hardware and software tools.
- To familiarize students with electrical safety practices, domestic energy calculations, and power measurement techniques.
- To introduce AI-assisted and open-source simulation tools for electrical circuit analysis, monitoring, visualization, and intelligent data interpretation.
- To enhance problem-solving, analytical, and experimental skills through AI-integrated laboratory practices in basic electrical engineering.

**Course Outcomes:**

**CO1:**UnderstandtheElectricalcircuitdesignconcept;measurementofresistance,power,power factor; conceptofwiringandoperation ofElectricalMachinesandTransformer

**CO2:** Apply the theoretical concepts and operating principles to derive mathematical models for circuits,Electricalmachinesandmeasuringinstruments;calculationsforthemeasurementof resistance, powerandpowerfactor

**CO3:** Apply the theoretical concepts to obtain calculations for the measurement of resistance, power and powerfactor

**CO4:**Analyzevariouscharacteristicsofelectricalcircuits,electricalmachinesandmeasuring instruments

**CO5:**Designsuitablecircuitsandmethodologiesforthemeasurementofvariouselectricalparameters; Householdandcommercialwiring

**List ofExperiments:**

1. VerificationofKCLandKVL
2. VerificationofSuperpositiontheorem
3. MeasurementofResistanceusingWheatstonebridge
4. MagnetizationCharacteristicsofDCshuntGenerator
5. MeasurementofPower andPower factorusingSingle-phasewattmeter
6. MeasurementofEarth ResistanceusingMegger
7. CalculationofElectrical EnergyforDomesticPremises

**AI-Integrated Topics**

AI-assisted circuit analysis and verification
Intelligent prediction of circuit responses using AI tools
AI-based resistance estimation and error analysis

AI-assisted prediction of magnetization characteristics
AI-based power factor monitoring and analysis
Intelligent earthing fault detection using AI techniques
AI-assisted domestic energy consumption and electricity bill prediction

**Suggested AI Tools**

LTspice
Scilab
KiCad + Ngspice
Python

**LearningResources:**

**Referencebooks:**

1. BasicElectricalEngineering,D.C.Kulshreshtha,TataMcGrawHill,2019,FirstEdition
2. PowerSystemEngineering,P.V.Gupta,M.L.Soni,U.S.BhatnagarandA.Chakrabar ti, Dhanpat Rai & Co, 2013
3. FundamentalsofElectricalEngineering,RajendraPrasad,PHIpublishers,2014,ThirdEdition.

## PART-B: Basic Electronics Engineering

**COURSE OBJECTIVES:**

- Introduce V-I and transfer characteristics of diodes and BJT and extract device parameters.
- Develop understanding of rectifier circuits, biasing networks, and amplifier design.
- Familiarize students with LLM-based circuit interpretation and ML techniques for parameter estimation.

**List of Experiments:** (Implement / execute any 06 experiments).

1. PlotV-IcharacteristicsofPNJunctionDiodeA) ForwardbiasB)Reversebias.
2. PlotV–IcharacteristicsofZenerDiodeanditsapplicationasvoltageRegulator.
3. Implementationofhalfwaveandfullwave rectifiers
4. PlotInput&OutputcharacteristicsofBJTinCEandCBconfigurations
5. FrequencyresponseofCEamplifier.
6. SimulationofRCcoupledamplifierwiththedesignsupplied
7. VerificationofTruthTableofAND,OR,NOT,NAND,NOR,Ex-OR,Ex-NORgatesusingICs.
8. VerificationofTruthTablesofS-R,J-K&DflipflopsusingrespectiveICs.

**AI/LLM and/or Python / MATLAB Tasks**

- Any three experiments need to be implemented using appropriate AI/LLM tools and/or Python/MATLABbased simulations.

1.a) Usean AI/LLM tool to interpret the forward bias, reverse bias, and breakdown regions of the V-I curve and explain the significance of threshold voltage and reverse saturation current.

b). Use ML/Python to Train a regression model to fit the diode equation (Shockley equation) to the measured V-I data and estimate saturation current (IS) and ideality factor (n).

2.a) Use an AI/LLM tool to explain the Zener breakdown mechanism, the difference between avalanche and Zener breakdown, and how the diode maintains voltage regulation under varying load.

b) Use ML/Python to Build a regression model to predict regulation performance (output voltage vs. load current) from measured characteristic data.

3.a) Use an AI/LLM tool to explain the working principle of half wave and full wave rectifiers, the role of the filter capacitor, and derive expressions for ripple factor and efficiency.

b) Use Python/MATLAB simulation to predict ripple voltage and efficiency. Apply regression to model ripple factor as a function of load resistance and capacitance value.

4.a) Use an AI/LLM tool to explain CE and CB input/output characteristics, the concept of current gain β and α, the Early effect, and compare the two configurations for impedance and gain.

b) Use ML/Python to Train a regression model on CE output characteristic data to estimate β and output resistance. Build a parameter prediction model for CB configuration to estimate α and input/output resistance.

5.a) Use an AI/LLM tool to analyse the BJT CE amplifier frequency response, explain low-frequency and high-frequency cutoff causes, and interpret the Bode plot.

b) Use ML/Python to curve fitting (polynomial or spline regression) to model the frequency response and estimate the -3 dB bandwidth automatically from simulation or measured data.

6.a) Use an AI/LLM tool to explain RC coupling, the purpose of coupling and bypass capacitors, inter-stage loading effects, and how these shape the frequency response.

b) Combine LLM-guided design with a regression/optimisation model to predict midband gain and bandwidth for different RC values, enabling design optimization

7.a) Use an AI/LLM tool to explain the Boolean algebra behind each gate, derive truth tables from first principles, and discuss NAND and NOR as universal gates with implementation examples.

b) Use ML/Python to Train a simple classifier (decision tree or logic network) using the truth table data as training set, and verify that the classifier reproduces correct outputs for all input combinations.

8.a) Use an AI/LLM tool to explain the operation of SR, JK, and D flip-flops, derive their characteristic equations, and explain why JK flip-flop eliminates the invalid state of SR flip-flop.

b) Use ML/Python to Train a sequence prediction model (e.g., simple RNN or state-table lookup) using flip-flop state transition data to predict Q(next) given current state Q and inputs; verify it matches theoretical characteristic equations.

.

**Instructions:**

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

## Course Outcomes:

***On successful completion of this lab, students will be able to:***

- Characterize electronic devices by plotting V-I curves and extracting parameters.
- Design and evaluate rectifier and amplifier circuits and measure key performance metrics.

- Apply ML regression and classifiers to estimate device parameters and automate analysis.

II Year I Sem

# ENGINEERING MECHANICS LAB

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3303P | B.Tech ME – II Year I Sem | 3 | 0 | 0 | 0 | 3 |

**Prerequisites:** Linear Algebra & Calculus, Engineering Physics
**Branch:** Mechanical Engineering (Inferred – not specified in the source.) | Regulation: JNTUA R26 (Inferred – assumed consistent with companion syllabus.)

## Course Objectives

- Validate foundational vector mechanics laws experimentally using standard laboratory hardware.
- Calculate essential material properties and physical constants including friction coefficients, gravitational acceleration, and moments of inertia.
- Train students in systematic data logging, error minimization, free-body diagram mapping, and graphical vector analysis.
- Determine the coefficients of static and rolling friction and the centre of gravity of different plane laminae.
- Analyse systems of pulleys and the moment of inertia of a compound pendulum and flywheel.

## List of Experiments

### MODULE: LIST OF EXPERIMENTS

- Verification of the Law of Parallelogram of Forces.
- Verification of the Law of Triangle of Forces.
- Verification of the Law of Polygon for coplanar-concurrent forces acting on a particle in equilibrium; find unknown forces using the universal force table.
- Determination of coefficient of static and rolling friction.
- Determination of the centre of gravity of different shaped plane laminae.
- Verification of the conditions of equilibrium of a rigid body under a coplanar non-concurrent, parallel force system using a simply supported beam.
- Study of systems of pulleys and drawing the free-body diagram of the system.
- Determination of the acceleration due to gravity using a compound pendulum.
- Determination of the moment of inertia of a compound pendulum about an axis perpendicular to the plane of oscillation, passing through its centre of mass.
- Determination of the moment of inertia of a flywheel.
- Verification of the Law of Moment using a Rotation Disc Apparatus and Bell Crank Lever.

## AI Integration & Industrial Applications

Incorporating automation and advanced computing bridges the gap between historical physics laws and modern automated engineering.

### AI-Based Computer Vision Tracking

- **Vector Force Mapping (Exps 1–3):** Students capture images of their balanced force systems or Universal Force Tables using a standard smartphone camera. A simple Python OpenCV convolutional neural network extracts string line vectors, measures the true angular intersections, and automatically calculates vector imbalances to instantly yield percentage errors.
- **Dynamic Oscillations & Motion Capture (Exps 8–10):** Instead of using manual stopwatches — which introduce human reaction delay — students record slow-motion videos of the flywheel or compound pendulum. Object detection networks track a coloured marker on the apparatus, logging the angular displacement over time to generate precise oscillation profiles.

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Verify the equilibrium conditions of coplanar concurrent systems via analytical, graphical, and physical methods. | L3 (Apply) |
| CO2 | Calculate reactions and load behaviours on simply supported rigid beams under non-concurrent parallel forces. | L3 (Apply) |
| CO3 | Determine the centre of gravity and moment of inertia of different configurations. | L3 (Apply) |
| CO4 | Differentiate between static and rolling friction thresholds over variable contact surfaces. | L4 (Analyze) |
| CO5 | Solve problems involving the translational and rotational motion of rigid bodies. | L3 (Apply) |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L3 | 3 | 3 | - | 1 | - | - | - | - | - | - | - | 1 |
| CO2 | L3 | 3 | 3 | 1 | 2 | - | - | - | - | - | - | - | 1 |
| CO3 | L3 | 3 | 3 | 2 | 2 | 1 | - | - | - | - | - | - | 2 |
| CO4 | L4 | 3 | 2 | 1 | 1 | - | - | - | - | - | - | - | 1 |
| CO5 | L3 | - | - | - | - | - | - | - | - | 3 | 3 | - | 2 |

## Textbooks

*(Suggested — not specified in the source.)*

- R. C. Hibbeler, Engineering Mechanics: Statics and Dynamics, Pearson Education.
- Ferdinand P. Beer & E. Russell Johnston Jr., Vector Mechanics for Engineers, McGraw Hill.

## Online Learning Resources

- Virtual Labs Repository – NPTEL & MHRD India Virtual Labs Portal, for online simulators focusing on Applied Mechanics and Vector Equilibrium setups.
- Video Tutorials – IIT Delhi Engineering Mechanics Lecture Series, hosted on NPTEL.

## Recommended Free & Open-Source Mechanics AI Tools

**Language & Environment:** Python, OpenCV – computer-vision-based force and motion analysis.

**Vision & Motion Capture:** Smartphone camera capture with object-detection networks – vector mapping and oscillation tracking.

**Visualization:** NPTEL Virtual Labs – online simulators for applied mechanics and vector equilibrium.

II Year I Sem

# MECHANICS OF SOLIDS & MATERIAL SCIENCE LAB

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3302P | I Year I Sem | 0 | 0 | 0 | 3 | 1.5 |

**Prerequisites:** Engineering Mechanics, Linear Algebra & Calculus
**Branch:** Common to Mechanical and Allied Branches (Inferred – not specified in the source.) | Regulation: JNTUA R26 (Inferred – assumed consistent with companion syllabus.)

## Course Objectives

- Evaluate the yield stress, ultimate stress, and bending stress of a given specimen under tension and bending tests.
- Conduct the torsion test to determine the modulus of rigidity of a given specimen.
- Measure the hardness of a given specimen using the Rockwell hardness test and the Brinell hardness test.
- Examine the stiffness of open-coil and closed-coil springs and grade them accordingly.
- Analyse the microstructure and characteristics of ferrous and non-ferrous alloy specimens.

## List of Experiments

**NOTE: Any 6 experiments from each of Section A and Section B are to be performed.**

### MODULE A: MECHANICS OF SOLIDS LAB

**Topics:** Tensile testing; bending of simply supported and cantilever beams; torsion testing; hardness testing (Brinell's, Rockwell, Vickers); testing of springs; impact testing (Charpy, Izod); punch shear testing; liquid penetration testing.

**AI Integration:** Use AI tools for automated UTM data acquisition, stress–strain curve analysis, and material property prediction.

**Suggested Tools:** Universal Testing Machine (UTM), Python/MATLAB, Digital Image Correlation (DIC)

**Experiments:**

- **Experiment 1: Tensile Test —** Determine yield stress, ultimate stress, and related mechanical properties of the given specimen.
- **Experiment 2: Bending Test —** Conduct bending tests on (a) a simply supported beam and (b) a cantilever beam.
- **Experiment 3: Torsion Test —** Determine the modulus of rigidity of the given specimen.
- **Experiment 4: Hardness Test —** Determine hardness using (a) Brinell's hardness test, (b) Rockwell hardness test, and (c) Vickers hardness test.
- **Experiment 5: Test on Springs —** Examine the stiffness of open-coil and closed-coil springs.
- **Experiment 6: Impact Test —** Determine impact resistance using (a) Charpy test and (b) Izod test.
- **Experiment 7: Punch Shear Test —** Determine the shear strength of the given specimen.
- **Experiment 8: Liquid Penetration Test —** Detect surface defects in the given specimen using liquid penetrant inspection.

### MODULE B: MATERIAL SCIENCE LAB

**Topics:** Microstructure of pure metals; mild, medium, and high carbon steels; cast irons; non-ferrous alloys; heat-treated steels; hardenability testing.

**AI Integration:** Use AI tools to classify microstructure images, generate automated test reports, and assist in interpreting metallurgical structures.

**Suggested Tools:** Metallurgical Microscope, Jominy End Quench Apparatus, Python/MATLAB

**Experiments:**

- **Experiment 1:** Preparation and study of the microstructure of pure metals.
- **Experiment 2:** Preparation and study of the microstructure of mild steel, medium carbon steel, and high carbon steel.

- **Experiment 3:** Study of the microstructures of cast irons.
- **Experiment 4:** Study of the microstructures of non-ferrous alloys.
- **Experiment 5:** Study of the microstructures of heat-treated steels.
- **Experiment 6:** Determination of the hardenability of steels by the Jominy End Quench Test.

## VIRTUAL LAB

- **1.** Investigate the principal stresses σa and σb at a given point of a structural element or machine component in a state of plane stress. (virtual-labs.github.io/exp-rockwell-hardness-experiment-iiith)
- **2.** Determine the impact resistance of mild steel and cast iron using the Izod impact test. (sm-nitk.vlabs.ac.in/exp/izod-impact-test)
- **3.** Determine the impact resistance of mild steel using the Charpy impact test. (sm-nitk.vlabs.ac.in/exp/charpy-impact-test)
- **4.** Determine the Rockwell hardness number of mild steel, cast iron, brass, aluminium, and spring steel. (sm-nitk.vlabs.ac.in/exp/rockwell-hardness-test)
- **5.** Determine the indentation hardness of mild steel, brass, and aluminium using a Vickers hardness testing machine. (sm-nitk.vlabs.ac.in/exp/vickers-hardness-test)

## HANDS-ON PRACTICES

- Universal Testing Machine (UTM) experiments.
- Stress–strain curve generation for different materials.
- Measurement of Poisson's ratio using strain gauges.
- Thermal expansion experiments.

## AI TOOLS INTEGRATION

- **Data Acquisition:** Automated UTM data collection.
- **Python/MATLAB:** Stress–strain curve analysis.
- **Machine Learning:** Material property prediction.
- **AI Vision:** Strain measurement using Digital Image Correlation (DIC).

## AI ACTIVITIES

- Build an AI system to classify materials from stress–strain curves.
- Create automated report generation for material testing.
- Use machine learning to predict failure points.
- Develop a virtual material-testing laboratory.

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Understand the stress–strain behaviour of different engineering materials. | L2 (Understand) |
| CO2 | Determine physical and mechanical properties (yield strength, ultimate tensile strength, hardness, and impact energy) of engineering materials. | L3 (Apply) |
| CO3 | Explain the relation between elastic constants and the hardness of materials. | L2 (Understand) |
| CO4 | Evaluate the hardness of different materials using standard hardness-testing methods. | L5 (Evaluate) |
| CO5 | Identify various microstructures of steels and cast irons. | L3 (Apply) |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L2 | 3 | - | - | 1 | 2 | - | - | - | - | - | - | 1 |
| CO2 | L3 | 3 | 3 | 2 | 2 | 1 | - | - | - | - | - | - | 1 |
| CO3 | L2 | 3 | 2 | 1 | 2 | 2 | - | - | - | - | - | - | 2 |
| CO4 | L5 | - | - | - | - | - | - | - | 1 | 3 | - | 1 | - |

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO5 | L3 | - | - | - | - | - | - | - | - | - | 3 | - | 1 |

## Textbooks

*(Suggested — not specified in the source.)*

- R. K. Bansal, Strength of Materials, Laxmi Publications.
- William D. Callister Jr. & David G. Rethwisch, Materials Science and Engineering: An Introduction, Wiley.

## Reference Books

- S. Timoshenko & James M. Gere, Mechanics of Materials, CBS Publishers.
- V. Raghavan, Materials Science and Engineering, PHI Learning.
- Sydney H. Avner, Introduction to Physical Metallurgy, McGraw Hill.

## Online Learning Resources

- Virtual Labs (IIIT-H) – Principal Stresses Experiment – virtual-labs.github.io
- Virtual Labs (NIT-K) – Izod Impact Test – sm-nitk.vlabs.ac.in
- Virtual Labs (NIT-K) – Charpy Impact Test – sm-nitk.vlabs.ac.in
- Virtual Labs (NIT-K) – Rockwell Hardness Test – sm-nitk.vlabs.ac.in
- Virtual Labs (NIT-K) – Vickers Hardness Test – sm-nitk.vlabs.ac.in
- NPTEL – Mechanics of Solids / Strength of Materials – nptel.ac.in

## Recommended Free & Open-Source Mechanics / Material Science AI Tools

**Language & Environment:** Python, MATLAB – automated data acquisition and stress–strain curve analysis.

**Testing & Measurement Tools:** Universal Testing Machine (UTM), Digital Image Correlation (DIC) – automated UTM data collection and AI-vision-based strain measurement.

**Machine Learning:** material property prediction and ML-based prediction of failure points from test data.

**AI Assistants:** automated report generation for material testing and development of a virtual material-testing laboratory.

II Year I Sem

# COMPUTER AIDED MACHINE DRAWING

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3304S | B. TECH-ME-II-I Sem | 1 | 0 | 0 | 4 | 3 |

**Prerequisites:** Engineering Graphics, Fundamental Python Programming
**Branch:** Mechanical Engineering (ME)

## Course Objectives

- Master standard drafting conventions, symbols, and geometric detailing for common fasteners and structural connections.
- Train to use software for 2D and 3D modelling.
- Familiarize with thread profiles, riveted, welded and key joints.
- Teach solid modeling of machine parts and their sections.
- Explain creation of 2D and 3D assembly drawings and Familiarize with limits, fits, and tolerances inmating components

## Course Content

### 2D Software Package Exercises

The following are to be done by any 2D software package

**Conventional representation of materials and components:**

**Detachable joints:** Drawing of thread profiles, hexagonal and square-headed bolts and nuts, bolted joint with washer and locknut, stud joint, screw joint and foundation bolts.

**Riveted joints:** Drawing of rivet, lap joint, butt joint with single strap, single riveted, double riveted double strap joints.

**Welded joints:** Lap joint and T joint with fillet, butt joint with conventions.

**Keys:** Taper key, sunk taper key, round key, saddle key, feather key, woodruff key.

**Couplings:** rigid – Muff, flange; flexible – bushed pin-type flange coupling, universal coupling, Oldham's coupling.

### 3D Software Package Exercises

The following exercises are to be done by any 3D software package:

**Sectional views:**
Creating solid models of complex machine parts and sectional views.

**Assembly drawings: (Any four of the following using solid model software)** Lathe tool post, tool head of shaping machine, tail-stock, machine vice, gate valve, carburetor, piston, connecting rod, eccentric, screw jack, plumber block, axle bearing, pipe vice, clamping device, Geneva cam, universal coupling.

**Production drawing:** Representation of limits, fits and tolerances for mating parts. Use any four parts of above assembly drawings and prepare manufacturing drawing with dimensional and geometric tolerances.

## Course Outcomes (COs)

*(Bloom's Level classifications are suggested; not specified in the source.)*

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Demonstrate the conventional representations of materials and machine components. | L2 (Understand) |
| CO2 | Model riveted, welded and key joints using CAD system. | L3 (Apply) |
| CO3 | Create solid models and sectional views of machine components. | L3 (Apply) |
| CO4 | Generate solid models of machine parts and assemble them. | L3 (Apply) |
| CO5 | Translate 3D assemblies into 2D drawings. | L3 (Apply) |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L2 | 3 | 2 | 2 | - | 1 | - | - | - | - | 2 | - | 2 |
| CO2 | L3 | 3 | 2 | 2 | - | 3 | - | - | - | - | 2 | - | 1 |
| CO3 | L3 | 3 | 2 | 3 | 1 | 3 | - | - | - | - | 1 | - | 2 |
| CO4 | L3 | 2 | 3 | 3 | 2 | 3 | - | - | - | 1 | 2 | - | 2 |
| CO5 | L3 | 2 | 2 | 3 | 1 | 3 | - | - | - | - | 3 | - | 2 |

## Textbooks

- Machine Drawing by K.L.Narayana, P.Kannaiah and K.Venkat Reddy, New Age International Publishers, 3/e, 2014
- Machine drawing by N.Sideshwar, P. Kannaiah, V.V.S.Sastry, TMH Publishers. 2014.

## Reference Books

- Cecil Jensen, Jay Helsel and Donald D.Voisinet, Computer Aided Engineering Drawing, Tata McGraw-Hill, NY, 2000.
- James Barclay, Brain Griffiths, Engineering Drawing for Manufacture, Kogan Page Science, 2003.
- N.D.Bhatt, Machine Drawing, Charotar Publishers, 50/e, 2014.

## Online Learning Resources

- https://eeedocs.wordpress.com/wp-content/uploads/2014/02/machinedrawing.pdf
- https://archive.nptel.ac.in/courses/112/105/112105294/
- https://www.edx.org/learn/engineering/dassault-systemes-solidworks-solidworks-cad-fundamentals?index=product&queryID=c90b35a82a6ef58b0d6f89679c63f6a1&position=2&linked_from=autocomplete&c=autocomplete
- https://www.youtube.com/watch?v=0bQkS3_3Fq4

II Year I Sem

| Course Code | ENVIRONMENTAL SCIENCE | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0301M | 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.<br>**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.<br>Free / Open-Source AI Tools: Google Earth Engine (free cloud GIS); QGIS (free, open-source); Python (scikit-learn, Matplotlib) – 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 ecosystem; Grassland ecosystem; Desert ecosystem; Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries).<br>**Biodiversity and its Conservation:** Introduction – 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 of biodiversity. |

| | |
|---|---|
| AI Integration | AI in Ecosystem and Biodiversity Studies: 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.<br>Free / Open-Source AI Tools: iNaturalist (free app/web); MaxEnt (free species distribution model); Google Earth Engine (free); QGIS (free); Python (rasterio, geopandas) – 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; Nuclear Hazards.<br>**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.<br>Free / Open-Source AI Tools: OpenAQ (free global AQ data API); QGIS (free); Python (scikit-learn, TensorFlow, Matplotlib) – 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).<br>Free / Open-Source AI Tools: Python (xarray, Matplotlib, scikit-learn) – 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.<br>**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).<br>Free / Open-Source AI Tools: iNaturalist (free); eBird (free); QGIS (free); Python (scikit-learn, TensorFlow) – Google Colab (free); OpenStreetMap (free). |

## Course Outcomes (COs)

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

| CO | Course Outcome Statement | Bloom's Taxonomy 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 (Remember, Understand) |
| **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 (Understand, Apply) |
| **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 (Apply, Analyze) |
| **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 (Analyze, Evaluate) |
| **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 (Understand, Apply) |

## Bloom's Taxonomy Coverage – Summary

| Bloom's Level | Level Name | CO(s) | Sample Activities / Assessment |
|---|---|---|---|

| | | | |
|---|---|---|---|
| **L1 – Remember** | Recall facts, define terms | **CO1, CO2** | Define ecosystem, biodiversity, pollution; state types of natural resources; name Indian environmental laws |
| **L2 – Understand** | Explain, describe, interpret | **CO1–CO5** | Explain ecological succession; describe food chain and food web; describe deforestation effects; explain climate change mechanisms |
| **L3 – Apply** | Use knowledge in new contexts | **CO2, CO3, CO5** | Apply SWM hierarchy to a local case; apply conservation strategies to a specific ecosystem; use iNaturalist/QGIS in field work |
| **L4 – Analyze** | Examine, compare, differentiate | **CO3, CO4** | Analyze pollution causes and effects for a given scenario; compare in-situ vs. ex-situ conservation; examine enforcement issues in environmental legislation |
| **L5 – Evaluate** | Assess, judge, critically evaluate | **CO4** | Evaluate the effectiveness of environmental laws in India; assess climate change mitigation policies; evaluate sustainability of a development project |
| **L6 – Create** | Design, propose (extension only) | **–** | Optional extension: propose an environmental management plan for a local site visited during field work |

## 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. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | L1-L2 | 2 | 1 | – | – | 1 | 3 | 3 | 2 | – | 2 |
| **CO2** | L2-L3 | 2 | 2 | – | 1 | 2 | 3 | 3 | 2 | 1 | 2 |
| **CO3** | L3-L4 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 1 | 2 |
| **CO4** | L4-L5 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 1 | 2 |
| **CO5** | L2-L3 | 1 | 1 | – | 1 | 3 | 3 | 3 | 2 | 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 & Online Platforms

- https://www.coursera.org/learn/python-for-applied-data-science-ai
- https://www.coursera.org/learn/python?specialization=python#syllabus
- 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 / Digital Tools |
| --- |
| Google Earth Engine (free cloud platform) – real-time satellite data analysis for deforestation, land use and vegetation change detection<br><br>QGIS (free, open-source GIS) – environmental mapping, watershed delineation, disaster risk zone mapping, biodiversity hotspot analysis<br><br>Python (NumPy, Pandas, Matplotlib, scikit-learn, TensorFlow) on Google Colab (free) – AQI prediction, population growth ML models, climate data analysis<br><br>iNaturalist (free, open-source citizen science) – species identification using image recognition during field work; community biodiversity database<br><br>eBird (free, Cornell Lab) – bird species recording and mapping during field visits; global bird distribution data<br><br>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<br><br>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

## BIOLOGY FOR ENGINEERS

**JNTUA R26 | II Year II Semester | Common to All Branches**

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AHS0304T | II Year II Sem | 2 | 0 | 0 | 0 | 2 |

*Prerequisites: Basic Sciences (Intermediate level)*

### Course Overview

This course introduces biological concepts through the lens of engineering and artificial intelligence, emphasising biomolecules, human systems, bio-inspired designs, and modern bioengineering trends, with practical case studies and AI-enabled problem-solving. It develops interdisciplinary thinking connecting biology, engineering, and computational tools.

### Course Objectives

- Provide foundational biological knowledge with engineering relevance.
- Demonstrate AI-driven applications in biology and bioengineering.
- Encourage interdisciplinary teaching that connects biology, engineering, and computational tools.
- Prepare students for innovation in healthcare, materials, and bio-industries.
- Evaluate biomimetic principles and emerging bioengineering trends for real-world engineering applications.

### Course Content

### UNIT I Foundations of Biology and AI

**Core Content:**

Cell as the basic unit of life. Carbohydrates, proteins, lipids, enzymes, vitamins, hormones — structure, function, and engineering relevance.

**AI Integration:**

AI-based visualization of biomolecules, Case studies on AI in drug discovery.

**Free/Open-Source AI Tools:**

AlphaFold (free via EBI); DeepChem (free, open-source); Python (NumPy, Matplotlib) on Google Colab (free); AI tools: ChatGPT / Gemini

### UNIT II Biomolecules in Engineering Applications

**Core Content:**

Carbohydrates in bioplastics (PHA, PLA). Proteins in food engineering (whey protein, plant analogs). Lipids in biodiesel and detergents. AI-driven generative design for biomaterials.

**AI Integration:**

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

**Free/Open-Source AI Tools:**

Python (NumPy, SciPy, Matplotlib) on Google Colab (free); ANSYS student version (free); AI tools: ChatGPT / Gemini

## UNIT III Human Systems and Bio-Designs

**Core Content:**

Brain as CPU (EEG, robotic prosthetics). Eye as camera (optical corrections, lens materials). Heart as pump (ECG, stents). Kidney as filtration (dialysis systems).

**AI Integration:**

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.

**Free/Open-Source AI Tools:**

Python (TensorFlow, scikit-learn, Matplotlib) on Google Colab (free); ANSYS student version (free); AI tools: ChatGPT / Gemini

## UNIT IV Bio-Inspired Materials and Mechanisms

**Core Content:**

Photosynthesis → photovoltaic cells. Bird flight → GPS and aircraft navigation. Lotus leaf → superhydrophobic surfaces. Kingfisher beak → bullet train design.

**AI Integration:**

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.

**Free/Open-Source AI Tools:**

Python (NumPy, SciPy, Matplotlib) on Google Colab (free); ANSYS student version (free); AI tools: ChatGPT / Gemini

## UNIT V Emerging Trends in Bioengineering

**Core Content:**

Muscular and skeletal scaffolds. 3D bioprinting of tissues (ear, bone, skin). AI in medical imaging (MRI, CT scans). AI in diagnostics and predictive healthcare.

**AI Integration:**

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

**Free/Open-Source AI Tools:**

Python (TensorFlow, OpenCV, Matplotlib) on Google Colab (free); ANSYS student version (free); AlphaFold (free via EBI); AI tools: ChatGPT / Gemini

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Explain the fundamentals of biology, biomolecules, and human systems relevant to engineering applications. | L2 (Understand) |
| CO2 | Apply AI tools and computational techniques for biological data analysis, biomolecule visualisation, and healthcare applications. | L3 (Apply) |
| CO3 | Analyse biomolecules and bio-inspired systems for solving engineering problems and developing sustainable technologies. | L4 (Analyze) |
| CO4 | Design biomimetic and bioengineering solutions using interdisciplinary approaches and AI-enabled tools. | L5 (Evaluate) L6 (Create) |
| CO5 | Evaluate emerging trends in bioengineering such as 3D bioprinting, regenerative medicine, and AI-based diagnostics for societal applications. | 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 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L2 | 3 | 2 | 1 | 1 | 1 | – | – | – | – | – | – | 2 | 2 | 1 |
| CO2 | L3 | 2 | 3 | 2 | 2 | 3 | – | – | – | 1 | – | – | 2 | 2 | 2 |
| CO3 | L4 | 2 | 3 | 3 | 2 | 2 | 2 | 3 | – | – | – | – | 2 | 2 | 2 |
| CO4 | L5,L6 | 2 | 2 | 3 | 2 | 3 | 2 | 2 | 1 | 2 | 1 | – | 2 | 2 | 2 |
| CO5 | L5 | 2 | 2 | 2 | 2 | 3 | 3 | 2 | 1 | 1 | 1 | – | 3 | 2 | 2 |

## Textbooks

1. Singh R.C. and Rao N.R., Biology for Engineers, Rajendra Singh C and Rathnakar Rao N Publishing, Bengaluru, 2023.
2. Thyagarajan S. et al., Biology for Engineers, Tata McGraw-Hill, New Delhi, 2012.

## Reference Books

1. Fox S. and Rompolski K., Human Physiology, McGraw-Hill eBook, 16th Edition, 2022.
2. Bar-Cohen Y., Biomimetics: Nature-Based Innovation, 1st Edition, CRC Press, 2012.
3. Ozbolat I., 3D Bioprinting: Fundamentals, Principles and Applications, Academic Press, 2016.

## Web Resources

- NPTEL Biology for Engineers – https://nptel.ac.in/courses/121106008
- MIT OpenCourseWare – Introduction to Biological Engineering: https://ocw.mit.edu
- AlphaFold Structure Database – https://alphafold.ebi.ac.uk

II Year II Sem

| Course Code | SUSTAINABILITY AND GREEN ENGINEERING | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3401T | | 2 | 0 | 0 | 0 | 2 |

## 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) – 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) – 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) – 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) – 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 (Remember) L2 (Understand) |
| 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 (Apply) |
| 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 (Analyze) |
| 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 (Analyze) L5 (Evaluate) |
| 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 (Evaluate) L6 (Create) |

## CO – PO Articulation Matrix

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

| CO | Bloom's Level | PO1 Eng. Knowledge | PO2 Problem Analysis | PO3 Design/ Dev. | PO4 Investigation | PO5 Modern Tools | PO6 Engineer & Society | PO7 Environ. & Sustain. | PO8 Ethics | PO9 Team Work | PO10 Commun. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L1, L2 | 2 | 1 | - | - | - | 3 | 3 | 2 | - | 2 |
| CO2 | L3 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 2 | 1 | 2 |
| CO3 | L4 | 2 | 2 | 2 | 2 | 3 | 2 | 3 | 1 | 1 | 2 |
| CO4 | L4, L5 | 2 | 2 | 3 | 2 | 3 | 3 | 3 | 2 | 1 | 2 |
| CO5 | L5, L6 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 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 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

# FLUID MECHANICS AND HYDRAULIC MACHINERY

| Course Code | Year/Sem | L T Pr P Credits |
|---|---|---|
| 26AME3402T | II Year II Sem | 2 0 2 0 3 |

**Prerequisites:** Engineering Mechanics, Linear Algebra & Calculus

## Course Objectives

- Teach fundamental physical properties of fluids and their engineering significance.
- Understand the kinematic and dynamic behaviour of fluids through the continuity, Euler's, Bernoulli's, energy, and momentum equations.
- Understand the theory of the boundary layer and the working and performance characteristics of hydraulic machines such as pumps and turbines.
- Teach impulse-momentum principles to evaluate hydrodynamic forces exerted by fluid jets on stationary and moving vanes.
- Teach the principles of geometric similarity and scale modeling for turbomachinery using unit and specific parameters.

## Course Content

### UNIT I: Fluid Statics, Buoyancy and Floatation

**Core Syllabus:** Fluid properties: dimensions and units; mass density, specific weight, specific volume, specific gravity; viscosity, Newton's law of viscosity, significance; surface tension, capillarity, vapour pressure, cavitation. Pressure measurement: fluid pressure at a point, Pascal's law, hydrostatic law; atmospheric, gauge, vacuum, and absolute pressure; piezometer, U-tube, inverted, and differential manometers. Buoyancy and floatation: buoyant force, principle of floatation, submerged and floating bodies, metacentre and metacentric height, analytical and experimental determination of metacentric height, stability criteria (stable, unstable, neutral equilibrium), engineering applications.

**Hands-on Practices:**

- Viscosity lab: measuring fluid viscosity using a Redwood or Saybolt viscometer at varying temperatures
- Manometer calibration: calibrating a Bourdon tube pressure gauge against a dead-weight tester and U-tube manometer
- Metacentric height experiment: determining the metacentric height of a ship model and analysing stability by moving weights
- Hydrostatic force bench: measuring the hydrostatic force exerted by a fluid on a submerged flat plate

**AI Integration:** AI models that predict changes in fluid viscosity across extreme temperature variations. Computer-vision-based smart manometry that reads and analyses physical manometer mercury levels automatically. Genetic algorithms that optimise ship hull geometry for maximum buoyant stability. A prompt-engineering exercise has students write a Python script that uses sensor data (tilt angle and lateral centre-of-gravity shift) to recalculate metacentric height and flag entry into the neutral or unstable stability zone.

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, open-source) on Google Colab for viscosity-prediction and metacentric-height calculators; OpenCV (free, open-source) for computer-vision manometer reading; open genetic-algorithm libraries (e.g. DEAP, open-source) for hull-optimisation exercises

### UNIT II: Fluid Kinematics, Dynamics & Closed Conduit Flow

**Core Syllabus:** Fluid kinematics: flow classification (steady/unsteady, uniform/non-uniform, laminar/turbulent, rotational/irrotational); continuity equation in 1D; circulation and vorticity; streamlines, pathlines, streaklines, and stream tubes; stream function and velocity potential function; flow nets, source, sink, doublet, and vortex flow. Fluid dynamics: surface and body forces; Euler's equation; Bernoulli's equation derivation and limitations; momentum equation application; forces acting on pipe bends. Closed conduit flow: Reynolds experiment and flow regimes; Darcy-Weisbach equation; friction factor in pipes; minor losses (expansion, contraction, bends); pipes in series and parallel; Total Energy Line (TEL) and Hydraulic Gradient Line (HGL).

**Hands-on Practices:**

- Flow-visualization lab: observing streamlines, streaklines, and laminar-to-turbulent transition using a Reynolds apparatus
- Bernoulli's theorem verification: validating energy conservation along a convergent-divergent duct
- Pipe friction experiment: measuring major head loss due to friction across pipe materials and diameters
- Minor losses apparatus: determining loss coefficients for sudden expansion, contraction, bends, and valves

**AI Integration:** Physics-informed neural networks (PINNs) that solve the Navier-Stokes equations by embedding continuity and Bernoulli constraints directly into the loss function. Smart leak detection using ML models that analyse pressure drops and HGL anomalies to pinpoint pipe-network leaks. Generative AI models that synthesise complex flow-net patterns for irregular engineering geometries. A prompt-engineering exercise has students write a Python (pandas) script to determine whether a sudden HGL drop at a network node matches a minor valve loss or indicates a structural pipe rupture.

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, pandas, open-source) on Google Colab for HGL and leak-detection analysis; PyTorch/TensorFlow (open-source) for physics-informed neural network solvers; open CFD tools (e.g. OpenFOAM) for flow-net visualisation

## UNIT III: Boundary Layer Theory & Dimensional Analysis

**Core Syllabus:** Boundary layer theory: concept of a boundary layer and its growth over a flat plate; Von Kármán momentum integral equation derivation; boundary layer thicknesses — displacement, momentum, and energy; velocity profiles — linear, parabolic, and logarithmic; boundary layer separation — physical causes, criteria, and engineering control methods. Dimensional analysis: fundamental and derived dimensions; principle of dimensional homogeneity; non-dimensionalization of differential governing equations; Buckingham Pi theorem.

**Hands-on Practices:**

- Wind-tunnel velocity profiling: mapping boundary layer thickness over a flat plate with a miniature Pitot-static tube
- Drag coefficient assessment: comparing pressure drag on a bluff sphere versus a streamlined teardrop airfoil model
- Flow separation tracking: visualising wake zones and vortex shedding using dye injection in a water-channel loop
- Scale model dimensioning lab: applying the Buckingham Pi theorem to design a scale model with water-glycerol similarity balancing

**AI Integration:** ML regression models that map near-wall turbulence and predict skin-friction drag coefficients directly from coarse grid data. Deep reinforcement learning that optimises real-time pulsing of active synthetic jets to delay aerodynamic flow separation. Large-language-model-assisted parsing of geometric variables to automatically generate complete, independent non-dimensional parameter spaces for novel multi-phase hydraulic systems. A prompt-engineering exercise has students write a Python (SciPy) script to numerically evaluate displacement and momentum thickness from a velocity dataset and classify boundary-layer state against the Blasius shape factor.

**Free/Open-Source AI Tools:** Python (SciPy, NumPy, scikit-learn, open-source) on Google Colab for boundary-layer-thickness and drag-coefficient calculations; open reinforcement-learning frameworks (PyTorch, open-source) for flow-separation control studies

## UNIT IV: Basics of Turbomachinery & Hydraulic Turbines

**Core Syllabus:** Basics of turbomachinery: hydrodynamic force of fluid jets; impulse-momentum principle; force exerted on stationary and moving flat, inclined, and curved vanes; jet striking centrally and at the tip; inlet and outlet velocity diagrams; derivation of work done, power, and efficiency; fluid flow over radial vanes. Hydraulic turbines: classification criteria (high, medium, low head; impulse vs. reaction); Pelton wheel — component design, velocity triangles, work done, maximum hydraulic efficiency, working proportions; Francis turbine — component layout, guide vanes, runner design, velocity triangles, work done, efficiencies; Kaplan turbine — axial-flow mechanics, adjustable blade proportions, performance parameters. Draft tube theory: types of draft tubes (conical, elbow), functions, pressure recovery, and efficiency.

**Hands-on Practices:**

- Impact of jets apparatus: comparing measured impact force of a water jet on flat and hemispherical vanes with theoretical momentum equations
- Pelton wheel performance: mapping brake power, discharge, and overall efficiency against rotational speed under constant head
- Francis turbine bench: load tests to plot operating characteristics and determine optimal guide-vane openings
- Draft tube simulation: mapping pressure distribution to observe kinetic-to-pressure-head energy conversion

**AI Integration:** ML neural networks that predict optimal runner blade angles instantaneously from fluctuating inlet flow parameters without iterative trigonometry. AI-driven cavitation diagnostics that combine acoustic sensor data with deep-learning classification to detect micro-bubble implosions in draft tubes before structural damage occurs. Reinforcement-learning agents that modify the 3D surface profiles of Francis runner blades to optimise part-load efficiency. A prompt-engineering exercise has students write a Python (NumPy) script that adjusts nozzle needle-valve position and recalculates the inlet velocity triangle to compensate for bucket wear while maintaining constant electrical output.

**Free/Open-Source AI Tools:** Python (NumPy, scikit-learn, open-source) on Google Colab for velocity-triangle and efficiency calculators; open deep-learning frameworks (PyTorch/TensorFlow, open-source) for cavitation-diagnostics and blade-optimisation models

## UNIT V: Performance of Hydraulic Turbines, Pumps & Fluidic Systems

**Core Syllabus:** Performance of hydraulic turbines: concept of geometric similarity; unit quantities — unit speed, unit discharge, unit power; specific quantities — specific speed derivation and significance; characteristic curves — main, operating, and constant-efficiency curves; governing of turbines — methods and mechanisms for impulse and reaction types; selection criteria for turbines; transient phenomena — cavitation mechanics, water-hammer equations, and surge-tank design/functions. Hydraulic systems and fluidics: working mechanisms and industrial applications of hydraulic ram, hydraulic lift, and hydraulic coupling; fluidics core concepts — fluidic amplifiers, fluidic sensors, and fluidic oscillators, relative advantages, structural limitations, and target engineering applications. Centrifugal pumps: classification and components, operating mechanism, expression for work done by impeller, head definitions (manometric head, static head, losses, and system efficiencies), specific-speed formulas, operational configurations (series and parallel), performance characteristic curves, NPSH calculation and cavitation minimisation. Reciprocating pumps: classification and working components, discharge equations, work done and power consumption, volumetric efficiency, slip and negative slip concepts, indicator diagrams (ideal diagram, distortion due to acceleration and friction), air-vessel functionality.

**Hands-on Practices:**

- Centrifugal pump performance mapping: plotting Total Head, Brake Power, and Efficiency against discharge at constant speed
- Pumps in series and parallel configurations: logging performance shifts to validate head and flow amplification rules
- Reciprocating pump fault analysis: generating indicator diagrams to estimate percentage slip and frictional pressure drops

- Turbine scaling virtual lab: applying Buckingham-Pi similarity invariants to project scale-model data onto a full-scale turbine

**AI Integration:** Deep-learning LSTM networks that process high-frequency penstock transducer feeds to predict and isolate water-hammer waves before pipe fatigue failure. Artificial neural networks that dynamically model fluid temperature, vapour pressure, and suction-manifold geometry to regulate variable-frequency drives (VFDs) and avoid pump cavitation (smart NPSH optimisation). Computer-vision models that analyse high-speed optical fluid patterns in micro-fluidic amplifiers to autonomously calibrate fluidic-oscillator feedback-loop switching delays.

**Free/Open-Source AI Tools:** Python (NumPy, pandas, open-source) on Google Colab for pump/turbine performance and NPSH calculators; PyTorch/TensorFlow (open-source) for LSTM-based water-hammer prediction; OpenCV (free, open-source) for fluidic-oscillator machine vision

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level* |
|---|---|---|
| CO1 | Solve complex pressure measurement problems using various types of manometers. | L3 (Apply) |
| CO2 | Evaluate flow fields using velocity potential, stream functions, and flow nets. | L5 (Evaluate) |
| CO3 | Apply boundary layer theory, flow separation, and dimensional analysis. | L3 (Apply) |
| CO4 | Estimate the hydrodynamic forces of a jet on vanes in different positions. | L3 (Apply) / L4 (Analyze) |
| CO5 | Understand the working principles and performance evaluation of hydraulic pumps and turbines. | L2 (Understand) |

**Bloom's levels were not specified in the source syllabus and have been inferred here from the verbs used in each outcome statement; please verify/adjust as needed.*

## CO – PO Articulation Matrix

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

*Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. PSO columns were not present in the source CO-PO matrix and have been omitted; add them here if the department defines PSOs for this programme.*

## Textbooks

1. Y.A. Cengel and J.M. Cimbala, Fluid Mechanics: Fundamentals and Applications, 6th Edition, McGraw Hill Publications, 2019.

2. Dixon, Fluid Mechanics and Thermodynamics of Turbomachinery, 7th Edition, Elsevier Publishers, 2014.

## Reference Books

1. P N Modi and S M Seth, Hydraulics & Fluid Mechanics Including Hydraulic Machines, Standard Book House, 2017.

2. R.K. Bansal, Fluid Mechanics and Hydraulic Machines, 10th Edition, Laxmi Publications (P) Ltd, 2019.

3. Rajput, Fluid Mechanics and Hydraulic Machines, S Chand & Company, 2016.

4. D.S. Kumar, Fluid Mechanics and Fluid Power Engineering, S K Kataria & Sons, 2013.

5. D. Rama Durgaiah, Fluid Mechanics and Machinery, 1st Edition, New Age International, 2002.

## Web Resources & Video Links

- NPTEL (IIT Kharagpur) – Fluid Mechanics: Introduction and Fluid Properties, Prof. Suman Chakraborty
- Khan Academy – Fluids: Density, Pressure, and Pascal's Principle
- MIT OpenCourseWare – Hydrostatics and Stability of Floating Bodies
- NPTEL (IIT Madras) – Advanced Fluid Mechanics: Fluid Kinematics and Dynamics, Prof. T.I. Eldho
- SimScale Academy – Computational Fluid Dynamics (CFD) basics for pipe networks and flow tracking
- NPTEL (IIT Kanpur) – Boundary Layer Theory and Applications, Prof. S. Chakraborty

## Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, pandas, scikit-learn) – free on Google Colab for fluid-property, HGL, boundary-layer, and pump/turbine calculators
- PyTorch/TensorFlow (free, open-source) – PINNs, LSTM water-hammer prediction, and cavitation-diagnostics models
- OpenCV (free, open-source) – manometer reading and fluidic-oscillator machine vision
- OpenFOAM (free, open-source) – flow-net and pipe-network CFD visualisation in place of proprietary CFD packages

II Year II Sem

# MANUFACTURING PROCESSES

| Course Code | Year/Sem | L T Pr P Credits |
|---|---|---|
| 26AME3403T | II Year II Sem | 2 0 2 0 3 |

**Prerequisites:** Engineering Workshop, Engineering Materials

## Course Objectives

- Teach the working principle of different metal casting processes and gating systems.
- Explain the working of different types of welding processes and welding defects.
- Know the nature of plastic deformation, cold and hot working processes, working of a rolling mill and its types, and extrusion processes.
- Understand the principles of forging, tools and dies, and the working of forging processes.
- Know about additive manufacturing.

## Course Content

### UNIT I: Metal Casting Processes and AI Defect Prediction

**Core Syllabus:** Casting: introduction to manufacturing patterns, steps involved in making a casting, types, allowances, materials. Patterns and pattern making: types of patterns, materials used for patterns, pattern allowances and their construction; moulding materials, properties, sand testing. Gating system design: risers, functions, types, placement. Methods of melting and types of furnaces; solidification of castings and casting defects — causes and remedies. Basic principles and applications of special casting processes: centrifugal, die, investment, and continuous casting; casting defects, causes and remedies.

**Hands-on Practices:**

- Pattern making using wood
- Sand mold preparation
- Sand testing for permeability and strength
- Metal pouring and casting
- Defect identification in castings
- Gating system design exercises
- Riser and feeder calculation

**AI Integration:** Intelligent pattern designer that takes final part dimensions and generates all allowances plus a 3D pattern model considering material, process, and orientation. Casting-simulation analyzer that models metal flow and solidification, identifies potential defects such as porosity and shrinkage, and suggests gating/riser modifications. CNN-based defect classifier trained on casting-defect images to identify blow holes, inclusions, and cracks and recommend corrective actions. Process selector that recommends the optimal casting process and compares cost, quality, and lead time from part geometry, material, and quantity, and a virtual foundry for interactive process simulation and safety training.

**Free/Open-Source AI Tools:** Python (scikit-learn/TensorFlow, open-source) on Google Colab for defect classification and process-parameter prediction; OpenCV (free, open-source) for casting-inspection computer vision; open-source CAD tools for pattern design (in place of proprietary MAGMA/ProCAST simulation packages)

### UNIT II: Metal Joining Processes

**Core Syllabus:** Welding: classification of welding processes, types of welded joints and their characteristics; gas welding, different types of flames and uses, oxy-acetylene gas cutting; basic principles of arc welding, power characteristics, manual metal arc welding, submerged arc welding, TIG and MIG welding, electro-slag welding. Resistance welding, friction welding, friction stir welding, forge welding, explosive welding, thermit welding, plasma arc welding, laser welding, electron beam welding, soldering and brazing. Heat-affected zones in welding, pre- and post-heating, welding defects — causes and remedies.

**Hands-on Practices:**

- Gas welding practice and arc welding of different joints
- MIG/TIG welding demonstrations
- Spot welding exercises
- Weld defect identification
- NDT methods demonstration and weld strength testing

**AI Integration:** Intelligent weld inspector where computer vision analyses weld images to detect cracks, porosity, and undercut, classify defect severity, and recommend accept/reject/repair. Welding-parameter advisor that recommends process, current, voltage, and speed and predicts penetration and heat-affected zone from a material database and ML model. AR-based welding trainer with real-time technique feedback and gamified bead-quality scoring. Automated weld-planning system that analyses a CAD model, plans welding sequence to minimise distortion, and generates a robot program, and a weld-quality predictor that monitors current/voltage/travel speed to predict tensile strength and fatigue life in real time.

**Free/Open-Source AI Tools:** Python (scikit-learn/TensorFlow, open-source) on Google Colab for weld-quality prediction; OpenCV (free, open-source) for weld defect detection; open IoT platforms for real-time weld monitoring (in place of proprietary Visual Environment/SYSWELD simulation software)

## UNIT III: Metal Forming Processes

**Core Syllabus:** Bulk forming: plastic deformation in metals and alloys — recovery, recrystallization, and grain growth; hot working and cold working, strain hardening and annealing. Bulk forming processes: forging — types of forging, forging defects and remedies; rolling — fundamentals, types of rolling mills and products, forces in rolling and power requirements; extrusion and its characteristics, types of extrusion, impact extrusion, hydrostatic extrusion; wire drawing and tube drawing.

**Hands-on Practices:**

- Hand forging operations and rolling experiments on a small mill
- Sheet metal cutting and bending
- Deep drawing trials and die design exercises
- Force and energy calculations
- Microstructure analysis of worked metals

**AI Integration:** Forming-process simulator that models forging, rolling, and extrusion, visualises material flow and strain distribution, predicts defects such as laps and cracks, and optimises die design and process parameters. Intelligent die designer that generates a die design from part geometry, material, tolerances, and production volume, then simulates and validates it. Force-and-energy calculator that selects the appropriate model and recommends equipment capacity. Spring-back compensator (ML model trained on forming data) that predicts spring-back and adjusts die geometry, and a computer-vision quality-inspection system for surface defects and dimensional accuracy.

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, scikit-learn, open-source) on Google Colab for force/spring-back calculators and prediction models; OpenCV (free, open-source) for surface-defect inspection (in place of proprietary DEFORM/Forge FEA software)

## UNIT IV: Sheet Metal Working, Powder Metallurgy, and Algorithmic Nesting

**Core Syllabus:** Sheet metal forming: blanking and piercing, forces and power requirement in these operations, deep drawing, stretch forming, bending, spring-back and its remedies, coining, spinning, types of presses and press tools. High-energy-rate forming processes: principles of explosive forming, electromagnetic forming, electrohydraulic forming, rubber-pad forming — advantages and limitations.

**Hands-on Practices:**

- Project: a Python optimisation script using a Genetic Algorithm library that accepts random flat-cut shapes (washers, bracket outlines) and iteratively arranges them inside a fixed 2D canvas boundary to maximise raw-sheet utilisation

- Application: automated nest-configuration engines used inside CNC laser and turret-punch production stations

**AI Integration:** Reinforcement-learning and heuristic genetic algorithms for high-density sheet-metal nesting layout planning, and machine-learning regression models to predict spring-back angles in complex multi-axis bending profiles.

**Free/Open-Source AI Tools:** Python (DEAP or similar genetic-algorithm libraries, open-source) on Google Colab for nesting optimisation; scikit-learn (free, open-source) for spring-back regression models

## UNIT V: Processing of Plastics, Ceramics, and Additive Manufacturing

**Core Syllabus:** Types of plastics (thermosetting and thermoplastics); injection molding, blow molding, extrusion molding, compression molding, transfer molding. Processing of ceramics: pressing, slip casting, sintering. Additive manufacturing: steps in additive manufacturing (AM), classification of AM processes, advantages of AM and types of materials for AM, VAT photopolymerization AM processes, extrusion-based AM processes, powder bed fusion AM processes, direct energy deposition AM processes, post-processing of AM parts, applications.

**Hands-on Practices:**

- CAD modelling for AM and STL file creation
- FDM printing with different orientations and infill
- Parameter study on layer height and temperature effects
- Topology optimisation exercises
- Post-processing, surface finishing, and quality measurement and testing

**AI Integration:** Neural network trained on G-code features (print time, support-volume percentage, maximum overhang angle, layer count, material type, environmental conditions) to predict print success or failure with pre-print validation. Camera-based real-time monitoring system using object-detection architectures (e.g. YOLO/Faster R-CNN) to flag layer-adhesion problems such as warping, under-extrusion, and layer shifts during FDM printing. Generative-design workflow that explores thousands of design variations under given loads and constraints, evaluates them via FEA, and presents Pareto-optimal lightweight designs. Digital twin of the complete FDM process chain — from STL file through slicing, thermal/warpage simulation, and post-processing — for virtual experimentation before physical printing.

**Free/Open-Source AI Tools:** Python (TensorFlow/PyTorch, open-source) on Google Colab for print-failure prediction and defect-detection models; OpenCV (free, open-source) for camera-based print monitoring; open-source slicers (e.g. Cura, PrusaSlicer) in place of proprietary Fusion 360/Altair Inspire generative-design tools

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level* |
|---|---|---|
| CO1 | Explain casting processes and design patterns and molds. | L2 (Understand) |
| CO2 | Apply welding and joining techniques for different materials. | L3 (Apply) |
| CO3 | Analyze metal forming processes and their applications. | L4 (Analyze) |
| CO4 | Understand sheet metal forming processes. | L2 (Understand) |
| CO5 | Learn about the different types of additive manufacturing processes. | L1 (Remember) / L2 (Understand) |

**Bloom's levels were not specified in the source syllabus and have been inferred here from the verbs used in each outcome statement; please verify/adjust as needed.*

## CO – PO Articulation Matrix

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

*Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. PSO columns were not present in the source CO-PO matrix and have been omitted; add them here if the department defines PSOs for this programme.*

## Textbooks

1. Kalpakjian S and Steven R Schmid, Manufacturing Processes for Engineering Materials, 5th Edition, Pearson Publications, 2007.

2. P.N. Rao, Manufacturing Technology – Vol I, 5th Edition, McGraw Hill Education, 2018.

## Reference Books

1. A. Ghosh and A.K. Malik, Manufacturing Science, East West Press Pvt. Ltd, 2010.
2. Lindberg and Roy, Processes and Materials of Manufacture, 4th Edition, Prentice Hall India Learning Private Limited, 1990.
3. R.K. Jain, Production Technology, Khanna Publishers, 2022.
4. Sharma P.C., A Text Book of Production Technology, 8th Edition, S Chand Publishing, 2014.
5. H.S. Shaun, Manufacturing Processes, 1st Edition, Pearson Publishers, 2012.
6. WAJ Chapman, Workshop Technology, 5th Edition, CBS Publishers & Distributors Pvt. Ltd, 2001.
7. Hindustan Machine Tools, Production Technology, Tata McGraw Hill Publishers, 2017.
8. Ian Gibson, David W Rosen, Brent Stucker, Additive Manufacturing Technologies: 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing, 2nd Edition, Springer, 2015.

## Web Resources & Video Links

- NPTEL – Manufacturing Processes-I, IIT Kharagpur
- Industrial AI and Vision Systems in Manufacturing – Steve Brunton Lecture Series
- Scikit-Learn Regression Modelling and OpenCV Image Classification Manuals

## Recommended Free & Open-Source AI/Computational Tools

- Python with scikit-learn/TensorFlow/PyTorch (free, open-source) – defect classification, quality, and process-parameter prediction on Google Colab
- OpenCV (free, open-source) – casting, welding, forming, and print-monitoring computer vision
- Python genetic-algorithm libraries such as DEAP (free, open-source) – sheet-metal nesting optimisation
- Open-source slicers such as Cura and PrusaSlicer (free) – additive manufacturing slicing and parameter study
- Open-source CAD tools (e.g. FreeCAD) – pattern and die design in place of proprietary CAD/FEA/simulation packages

II Year II Sem

# KINEMATICS OF MACHINERY

| Course Code | Year/Sem | L T Pr P Credits |
|---|---|---|
| 26AME3404T | II Year II Sem | 2 0 2 0 3 |

**Prerequisites:** Engineering Mechanics, Linear Algebra & Calculus

## Course Objectives

- Analyze kinematic inversions derived from four-bar (quadric), single-slider, and double-slider crank chains.
- Explain exact and approximate straight-line motion mechanisms.
- Explain the concept of the instantaneous centre.
- Teach gears, gear trains, and their applications.
- Describe cams and followers and their motions.

## Course Content

### UNIT I: Mechanisms and Machines

**Core Syllabus:** Basics of mechanisms: definitions of link/element; classification of links — rigid, flexible, and fluid links; types of kinematic pairs — sliding, turning, rolling, screw, and spherical pairs; classification of pairs — lower vs. higher pairs, open vs. closed pairs. Types of constrained motion: completely constrained, partially (successfully) constrained, and incompletely constrained motions. Structure and mobility: difference between mechanisms and machines; classification of mechanisms and machines; concept of a kinematic chain; mobility of mechanisms — Grubler's criterion, Kutzbach criterion, and exceptions. Kinematic chain, inversion of mechanisms, inversions of the quadric cycle chain, single and double slider crank chain; mobility of mechanisms.

**Hands-on Practices:**

- Mechanism model assembly: reconstructing four-bar, single-slider, and double-slider mechanisms and mapping their dead-centre positions
- Mobility verification lab: graphically mapping physical assemblies and checking measured degrees of freedom against Kutzbach's criterion
- Quick-return motion analysis using a stroke-measuring rig on a crank-and-slotted-lever mechanism
- Oldham's coupling alignment tester: evaluating rotational-velocity transmission across intentionally misaligned shafts

**AI Integration:** CNN-based computer-vision pair classification that analyses real-time video of industrial robotic joints to classify link joints as turning, sliding, or spherical pairs. Generative synthesis of links using GANs or diffusion models to produce optimised, lightweight topology profiles for flexible links under variable loads. Automated mobility checkers that parse 2D engineering schematics, isolate nodes, and compute structural mobility to flag locked mechanism designs during CAD drafting. A prompt-engineering exercise has students write a Python script that applies the Kutzbach and Grubler criteria to a given link/pair schematic to classify a mechanism as locked, constrained, or unconstrained.

**Free/Open-Source AI Tools:** Python (NumPy, open-source) on Google Colab for mobility/DOF calculators; open-source CAD tools for schematic-based mobility checking; open GAN/diffusion frameworks (PyTorch, open-source) for link-topology synthesis

### UNIT II: Straight Line, Steering & Universal Joint Mechanisms

**Core Syllabus:** Straight-line motion mechanisms: definitions and classification — exact vs. approximate, copied vs. generated types; exact straight-line mechanisms (Peaucellier, Hart, Scott-Russell linkages); approximate straight-line mechanisms (Grasshopper, Watt, Tchebicheff, Robert mechanisms); geometry, working principles, and practical applications. Pantograph: kinematics of pantograph linkages, scaling principles, and path replication. Steering mechanisms: principles of correct steering; inner and outer wheel slip conditions; structural layouts and performance comparison of the Davis steering gear (sliding pairs) and the Ackermann steering gear (turning pairs). Hooke's joint (universal coupling): single and double Hooke's joint, applications, simple problems.

**Hands-on Practices:**

- Straight-line linkage mapping: fabricating scale models of Watt and Peaucellier linkages and tracking their physical paths with a stylus against a reference line
- Automotive steering geometry lab: measuring toe-in, toe-out, and turning radius on an Ackermann steering test rig
- Hooke's joint velocity profiling: charting angular-velocity fluctuations at various shaft angles using rotary encoders on a dual-shaft system

**AI Integration:** ML regression models that optimise link-length ratios for approximate straight-line mechanisms to minimise path deviation down to micron levels. Deep reinforcement learning that runs continuous kinematic simulations to dynamically tune Ackermann steering linkages for optimal tyre traction during cornering. Neural networks that process vibration and angular-speed data from universal joints to estimate bearing fatigue life. A prompt-engineering exercise has students write a Python (NumPy/Matplotlib) script to calculate and plot the angular-velocity ratio of a single Hooke's joint operating at a given shaft angle, labelling the maximum and minimum speed-fluctuation points.

**Free/Open-Source AI Tools:** Python (NumPy, SciPy, Matplotlib, scikit-learn — open-source) on Google Colab for path-synthesis optimisation and coupling-velocity calculations

## UNIT III: Velocity, Acceleration & Instantaneous Centre Analysis

**Core Syllabus:** Kinematics: motion of a rigid link in a plane. Relative velocity method: velocity polygons for four-bar and slider-crank mechanisms. Relative acceleration method: radial and tangential components of acceleration; acceleration polygons for simple mechanisms. Coriolis acceleration: concept, mathematical derivation, direction-determination rules, and application to mechanisms containing a slider on a rotating link (e.g. crank-and-slotted-lever mechanism). Klein's construction: graphical analysis of displacement, velocity, and acceleration of a slider-crank mechanism; analytical method. Instantaneous centre method: instantaneous centre of rotation, centrode and axode, relative motion between two bodies, three-centres-in-line theorem, locating instantaneous centres for simple mechanisms, and determination of angular velocity of points and links.

**Hands-on Practices:**

- Physical mechanism model construction
- Velocity and acceleration polygon construction
- Measurement using high-speed cameras
- Working models of four-bar and slider-crank mechanisms

**AI Integration:** Kinematic simulation using mechanism-design software such as SAM or GIM, and Python-based automated kinematic analysis (e.g. PyMechanisms). Computer-vision motion tracking from video and AR tools for mechanism visualisation. Activities include building a Python-based mechanism simulator, using ML to optimise linkage dimensions, developing an AR application for mechanism visualisation, and building an AI tool for automatic DOF calculation.

**Free/Open-Source AI Tools:** Python with PyMechanisms and NumPy/Matplotlib (open-source) on Google Colab for kinematic analysis and simulation; OpenCV (free, open-source) for video-based motion tracking

## UNIT IV: Gears & Gear Trains

**Core Syllabus:** Gears (toothed gearing): introduction to higher pairs; types of gears; law of gearing — condition for constant velocity-ratio transmission; forms of teeth — cycloidal and involute profiles, features, and profile comparison; kinematics of meshing — velocity of sliding; phenomena of interference and undercutting, methods to avoid interference; governing expressions — path of contact, arc of contact, contact ratio, and condition for the minimum number of teeth on a pinion/wheel to avoid interference. Gear trains: introduction and functional classification; performance analysis of simple, compound, reverted, and epicyclic gear trains; calculation of train value and velocity ratio; analysis of epicyclic gear trains using the tabular column method; torques in epicyclic gear trains; the differential gear of an automobile; simple numerical problems.

**Hands-on Practices:**

- Gear tooth profile measurement
- Velocity-ratio verification on gear models
- Epicyclic gear-train demonstration models
- Gear-cutting demonstrations

**AI Integration:** Comprehensive gear analysis using dedicated gear-design software (e.g. KISSsoft) and Python-based automation of gear-train calculations. Parametric gear modelling in CAD software and AI-based load-distribution simulation. Activities include creating an AI-powered gear-train calculator, using ML for gear-failure prediction, developing a generative-design tool for gear arrangements, and building a virtual gear-train design studio.

**Free/Open-Source AI Tools:** Python (NumPy, open-source) on Google Colab for gear-train and velocity-ratio calculators; open-source parametric CAD tools (e.g. FreeCAD) for gear modelling in place of proprietary KISSsoft

## UNIT V: Cams & Analysis of Follower Motion

**Core Syllabus:** Cams: definitions of cam and follower, uses, types of followers and cams, terminology. Types of follower motion — uniform velocity, simple harmonic motion, cycloidal, and uniform acceleration and retardation; maximum velocity and maximum acceleration during outward and return strokes; drawing of cam profiles. Analysis of motion of followers: tangent cam with roller follower; circular-arc (convex) cam with flat-faced and roller follower.

**Hands-on Practices:**

- Follower motion mapping rig: instrumenting an adjustable cam-follower bench with linear displacement sensors to distinguish SHM from cycloidal motion
- Cam profile CAD drafting studio: generating parameterised cam profiles in SolidWorks or Autodesk Inventor and checking for undercutting at tight radii
- Pressure-angle threshold evaluation: varying roller-follower offset to measure side-thrust friction and follower binding
- High-speed acceleration diagnostics: using an optical encoder and high-speed camera to capture acceleration spikes at tangent-cam transition points

**AI Integration:** ML regression models (AI follower-profile optimiser) that design custom, non-standard transition curves to bypass peak acceleration shocks for given speed limits. Physics-informed neural networks (jerk suppression) that model high-speed valve trains and optimise cam profiles to minimise high-frequency jerk and prevent spring surge. Deep-learning acoustic classifiers that parse vibration/acoustic time-series data to detect micro-wear on cam noses or follower scuffing before catastrophic clearance loss. A prompt-engineering exercise has students write a Python (NumPy/Matplotlib) script to calculate and plot the velocity and acceleration profile of a roller follower under SHM across the outstroke, flagging the angular location where acceleration changes sign.

**Free/Open-Source AI Tools:** Python (NumPy, Matplotlib, open-source) on Google Colab for follower-motion and acceleration calculations; open-source physics-informed neural network frameworks (PyTorch, open-source) for jerk-suppression modelling

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level* |
|---|---|---|
| CO1 | Identify the types of links, kinematic pairs, and relative constraints within any engineering mechanism. | L1 (Remember) / L2 (Understand) |
| CO2 | Understand the difference between Davis's and Ackermann's steering gear mechanisms. | L2 (Understand) |
| CO3 | Draw the velocity and acceleration diagrams for different mechanism configurations. | L3 (Apply) |
| CO4 | Gain knowledge of different gear profiles and calculate the various parameters of gears. | L2 (Understand) / L3 (Apply) |
| CO5 | Design and analyze different cam profiles for different types of followers. | L4 (Analyze) / L6 (Create) |

**Bloom's levels were not specified in the source syllabus and have been inferred here from the verbs used in each outcome statement; please verify/adjust as needed.*

## CO – PO Articulation Matrix

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

*Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. PSO columns were not present in the source CO-PO matrix and have been omitted; add them here if the department defines PSOs for this programme.*

## Textbooks

1. S.S. Rattan, Theory of Machines, Tata McGraw Hill Publishers.
2. J.E. Shigley, The Theory of Machines, McGraw Hill.

## Reference Books

1. R.S. Khurmi and J.K. Gupta, Theory of Machines, S. Chand Publications.
2. R.K. Bansal and J.S. Brar, Theory of Machines, Laxmi Publications.
3. Thomas Bevan, Theory of Machines, CBS.
4. J.S. Rao and R.V. Dukkipati, Mechanism and Machine Theory, New Age International.
5. R.L. Norton, Kinematics and Dynamics of Machinery, Tata McGraw Hill Publishers.

## Web Resources & Video Links

- NPTEL (IIT Kharagpur) – Kinematics of Mechanisms and Machines: Cam Dynamics and Profiles, Prof. A. Dasgupta
- MIT OpenCourseWare – Mechanical Elements: Kinematics and Structural Design of High-Speed Cam Systems
- NPTEL (IIT Madras) – Kinematics of Special Mechanisms: Straight-Line Linkages and Universal Couplings, Prof. Ashok Kumar Mallik
- MIT OpenCourseWare – Advanced Linkage Mechanics: Synthesis of Straight-Line Motions and Steering Geometry

## Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, Matplotlib, scikit-learn) – free on Google Colab for kinematic, gear, and cam calculations
- OpenCV (free, open-source) – motion tracking and computer-vision pair classification
- PyTorch (free, open-source) – GAN/diffusion link synthesis and physics-informed neural networks
- FreeCAD (free, open-source) – parametric gear and cam-profile modelling in place of proprietary KISSsoft/SolidWorks

II Year II Sem

# THERMODYNAMICS

| Course Code | Year/Sem | L T Pr P Credits |
|---|---|---|
| 26AME3405T | [Year/Sem not specified] | 2 0 2 0 3 |

**Prerequisites:** Engineering Physics and Linear Algebra & Calculus

## Course Objectives

- Familiarize concepts of heat, work, energy and governing rules for conversion of one form to other.
- Explain relationships between properties of matter and basic laws of thermodynamics.
- Teach the concept of entropy for identifying the disorder and feasibility of a thermodynamic process.
- Introduce the concept of available energy for maximum work conversion.
- Provide fundamental concepts of gas mixtures and gas power cycles and to have the understanding of basic power cycles.

## Course Content

### UNIT I: Basic Concepts and Zeroth Law of Thermodynamics

**Core Syllabus:** Introduction: Thermodynamic systems, types, macroscopic and microscopic approaches, state, property, process, cycle, thermodynamic equilibrium; Zeroth Law of Thermodynamics. Energy Transfers: work transfer, types of work transfers, point and path functions, heat transfer, comparison of work and heat transfers; numerical problems.

**Hands-on Practices:**

- Temperature measurement using different thermometers
- Pressure measurement using manometers and gauges
- Calibration of temperature sensors
- System identification exercises
- P-V diagram plotting manually
- Quasi-static process demonstration

**AI Integration:** Python-based thermodynamic property calculator covering unit conversions, property relationships, and a database of common substances. Interactive 3D P-V-T diagram explorer with process-path animation and state-point calculation. AI-based concept tutor (ChatGPT/Claude) for answering student queries and generating practice problems. Virtual measurement laboratory with simulated instruments and realistic sensor readings. ML-based process classifier to identify process type from P-V diagram images.

**Free/Open-Source AI Tools:** Python (NumPy/SciPy/Matplotlib) on Google Colab; PhET simulations (free virtual labs); AI tutors such as ChatGPT/Claude (free tier); open IoT data-logging tools for temperature/pressure sensors

### UNIT II: First Law of Thermodynamics

**Core Syllabus:** Energy in state and in transition, types, work and heat, point and path functions; Zeroth Law of Thermodynamics, PMM-I, Joule's experiment; First Law of Thermodynamics and applications; limitations of the First Law; enthalpy, thermal reservoir, heat engine, heat pump, and parameters of performance.

**Hands-on Practices:**

- P-V work measurement using piston-cylinder apparatus
- Heat transfer measurement using calorimeter
- Flow measurement in nozzles and venturi
- Temperature and pressure drop in throttling
- Pump and compressor performance testing

- Heat exchanger effectiveness measurement

**AI Integration:** AI-assisted Steady Flow Energy Equation (SFEE) analyzer that applies the appropriate form of SFEE to a given device and generates an energy-flow diagram. Process simulator for compression, expansion, and heating with real-time P-V-T calculation and energy-balance verification. AI-powered device analyzer that computes turbine/compressor efficiency from uploaded data and benchmarks it against industry standards. Virtual laboratory for SFEE applications (nozzle, turbine, compressor) with interactive inlet-condition variation. Adaptive problem generator that creates varied SFEE problems with complete worked solutions.

**Free/Open-Source AI Tools:** Python for process simulation and automated equation solving on Google Colab; open data-analytics libraries for experimental data analysis

## UNIT III: Second Law and Entropy

**Core Syllabus:** Second Law of Thermodynamics; Kelvin-Planck and Clausius statements and their equivalence/corollaries; PMM-II; Carnot's principle, Carnot cycle and its specialties; thermodynamic scale of temperature; Clausius inequality; entropy and the principle of entropy increase; energy equation; availability and irreversibility; thermodynamic potentials, Gibbs and Helmholtz functions, Maxwell relations; elementary treatment of the Third Law of Thermodynamics.

**Hands-on Practices:**

- Carnot cycle demonstration
- Reversibility vs irreversibility demonstrations
- Entropy calculations for various processes
- T-s diagram construction
- Heat engine efficiency experiments
- Refrigerator COP measurement

**AI Integration:** Interactive Carnot cycle simulator that animates the cycle on P-V and T-s diagrams while calculating work and heat transfer for adjustable temperatures. Entropy-change calculator that selects the appropriate formula for a given process and substance and generates a detailed solution. Reversibility analyzer that identifies irreversibilities in a described process and quantifies entropy generation. ML-based second-law efficiency optimizer for thermal systems under practical constraints. Animated concept-visualization tool with interactive diagrams and adaptive self-assessment quizzes.

**Free/Open-Source AI Tools:** Python for entropy calculators and numerical entropy integration on Google Colab; open cycle-simulation and animation tools for reversible-process visualization

## UNIT IV: Properties of Pure Substances

**Core Syllabus:** Pure substances, P-V-T surfaces, T-S and h-s diagrams, Mollier charts; phase transformations, triple point, critical-state properties during change of phase; dryness fraction; Clausius-Clapeyron equation; property tables; Mollier charts and various thermodynamic processes and energy transfer; steam calorimetry; real gas behaviour and compressibility factor.

**Hands-on Practices:**

- Steam table interpolation exercises
- Mollier diagram usage for process analysis
- Property determination experiments
- Quality measurement of wet steam
- Ideal vs real gas comparison
- Compressibility factor calculations

**AI Integration:** Intelligent property calculator that interprets natural-language queries (e.g. finding enthalpy of steam at a given pressure and temperature) and returns relevant additional properties. Interactive phase-diagram explorer that displays properties at any clicked point and animates phase transitions across substances. Automatic problem solver that identifies knowns/unknowns from a text problem statement, suggests a solution approach, and performs the calculation. Custom property-table generator for any substance and range, exportable in multiple formats. ML-based property predictor that interpolates/extrapolates when standard tables are unavailable.

**Free/Open-Source AI Tools:** Python open-source thermodynamic property libraries such as CoolProp and iapws (used in place of proprietary REFPROP) on Google Colab

## UNIT V: Power and Refrigeration Cycles

**Core Syllabus:** Introduction to refrigeration: working of air and vapour-compression systems, VCR system components, COP, refrigerants. Introduction to air conditioning: psychrometric properties and processes, characterisation of sensible and latent heat loads, load concepts of SHF; requirements of human comfort, concept of effective temperature, comfort chart, comfort air conditioning, and load calculations.

**Hands-on Practices:**

- Construction of P-H diagram and measurement of properties
- VCR plant components study
- Refrigeration cycle demonstration
- COP measurement of refrigerator
- Air conditioning cycle analysis

**AI Integration:** Comprehensive cycle analyzer that calculates all state points for a given cycle type and parameters, generates P-V and T-s diagrams, and computes efficiency/COP against actual data. Multi-objective cycle optimizer balancing efficiency, cost, and emissions with sensitivity analysis. Virtual Rankine-cycle power-plant simulator with adjustable operating conditions, real-time performance display, and fault diagnosis. Refrigeration-system selector that recommends cycle type, refrigerant, and component specifications from input cooling requirements. ML-based performance predictor trained on plant data for efficiency-degradation and maintenance scheduling.

**Free/Open-Source AI Tools:** Python open-source thermodynamic cycle libraries on Google Colab; open digital-twin/simulation frameworks for power-plant modelling (used in place of proprietary GATEWAY/PEACE/CFD packages)

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level* |
|---|---|---|
| CO1 | Explain the importance of thermodynamic properties related to conversion of heat energy into work. | L2 (Understand) |
| CO2 | Apply the First Law of Thermodynamics to non-flow and flow processes. | L3 (Apply) |
| CO3 | Understand the Second Law of Thermodynamics and entropy, and analyse reversible and irreversible processes. | L3 (Apply) / L4 (Analyze) |
| CO4 | Analyze the Mollier charts, T-S and h-s diagrams, steam calorimetry, and phase transformations. | L4 (Analyze) |
| CO5 | Understand the properties of gases and apply ideal/real gas equations of state. | L2 (Understand) / L3 (Apply) |

**Bloom's levels were not specified in the source syllabus and have been inferred here from the verbs used in each outcome statement; please verify/adjust as needed.*

## CO – PO Articulation Matrix

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

*Scale: 3 – Strong, 2 – Moderate, 1 – Slight, – No Correlation. PSO columns were not present in the source CO-PO matrix and have been omitted; add them here if the department defines PSOs for this programme.*

## Textbooks

1. P.K. Nag, Engineering Thermodynamics, 5th Edition, Tata McGraw Hill, 2013.

2. Claus Borgnakke and Richard E. Sonntag, Fundamentals of Thermodynamics, 7th Edition, Wiley, 2009.

## Reference Books

1. J.B. Jones and R.E. Dugan, Engineering Thermodynamics, 1st Edition, Prentice Hall, 1995.
2. Y.A. Cengel and M.A. Boles, Thermodynamics – An Engineering Approach, 7th Edition, McGraw Hill, 2010.
3. P. Chattopadhyay, Engineering Thermodynamics, 1st Edition, Oxford University Press, 2011.
4. C.P. Arora, Refrigeration and Air-conditioning, 4th Edition, McGraw Hill, 2021.

## Web Resources & Video Links

- NPTEL: Basic Thermodynamics – IIT Kharagpur: https://archive.nptel.ac.in/courses/112/106/112106310
- Physics-Informed Deep Learning (PINNs Tutorial Series, Steve Brunton, University of Washington)
- CoolProp and Python Wrapper Integration Guides
- edX – Thermodynamics: https://www.edx.org/learn/thermodynamics
- YouTube lecture: https://www.youtube.com/watch?v=7NI5P4KqrAs&t=1s
- Coursera – Introduction to Thermodynamics: https://www.coursera.org/learn/thermodynamics-intro

## Recommended Free & Open-Source AI/Computational Tools

- Python (NumPy, SciPy, Matplotlib) – free on Google Colab for process, property, and cycle calculations
- CoolProp / iapws (free, open-source) – thermodynamic property libraries for Python
- PhET Interactive Simulations (free) – virtual measurement and process labs
- AI tutors such as ChatGPT/Claude (free tier) – concept clarification and problem generation

II Year II Sem

# FLUID MECHANICS AND HYDRAULIC MACHINERY LAB

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3402P | II Year II Sem | 0 | 0 | 0 | 3 | 1.5 |

**Prerequisites:** Engineering Mechanics, Linear Algebra & Calculus
**Branch:** Common to Mechanical and Allied Branches (Inferred – not specified in the source.) | Regulation: JNTUA R26 (Inferred – assumed consistent with companion syllabus.)

## Course Objectives

- Introduce calibration procedures for flow measurement devices like venturimeters, orifice meters, and V-notches.
- Demonstrate validation techniques for Bernoulli's theorem, friction factors in pipelines, and minor losses.
- Train students to instrument hydraulic machinery (Pelton, Francis, Kaplan turbines).
- Explain the working of different types of pumps.
- Determine the coefficient of impact of a jet on vanes.

## List of Experiments

### MODULE: LIST OF EXPERIMENTS

- Impact of jets on vanes.
- Performance test on Pelton wheel.
- Performance test on Francis turbine.
- Performance test on Kaplan turbine.
- Performance test on single-stage centrifugal pump.
- Performance test on multi-stage centrifugal pump.
- Performance test on reciprocating pump.
- Calibration of venturimeter.
- Calibration of orifice meter.
- Determination of friction factor for a given pipeline.
- Determination of loss of head due to sudden contraction in a pipeline.
- Turbine flow meter.

### VIRTUAL LAB

- **1.** Study different patterns of flow through a pipe and correlate them with the Reynolds number of the flow. (me.iitp.ac.in/Virtual-Fluid-Laboratory/reynolds)
- **2.** Calculate the total energy at different points of a venturimeter. (me.iitp.ac.in/Virtual-Fluid-Laboratory/bernoulli)
- **3.** Calculate the flow (point) velocity at the centre of a given tube at different flow rates. (me.iitp.ac.in/Virtual-Fluid-Laboratory/pitot)
- **4.** Determine the hydrostatic force on a plane surface under partial and full submerged conditions. (me.iitp.ac.in/Virtual-Fluid-Laboratory/cop)
- **5.** Determine the discharge coefficient of a triangular notch. (me.iitp.ac.in/Virtual-Fluid-Laboratory/notch)
- **6.** Determine the coefficient of impact of a jet on vanes. (fm-nitk.vlabs.ac.in/exp/impact-of-jet)
- **7.** Determine friction in pipes. (fm-nitk.vlabs.ac.in/exp/friction-in-pipes)

## AI Integration Module

- Training Deep Neural Network (DNN) surrogates to instantly project turbine-efficiency topology surfaces; automated parameter tuning for optimal jet deflection.

- Acoustic and vibration time-series anomaly detection for rotating fluid machinery; deep classification frameworks detecting internal structural impeller cavitation.

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Calibrate flow meters and calculate exact discharge coefficients. | L3 (Apply) |
| CO2 | Demonstrate the devices used for measuring flow. | L2 (Understand) |
| CO3 | Compute major and minor losses in pipes. | L3 (Apply) |
| CO4 | Illustrate the operating parameters of turbines. | L3 (Apply) |
| CO5 | Explain the working of different types of pumps. | L2 (Understand) |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L3 | 3 | 2 | - | 1 | - | - | - | - | - | - | - | 1 |
| CO2 | L2 | 3 | 3 | 1 | 2 | - | - | - | - | - | - | - | 1 |
| CO3 | L3 | 3 | 3 | 2 | 2 | 1 | - | 1 | - | - | - | - | 2 |
| CO4 | L3 | 3 | 3 | 2 | 2 | 1 | - | 1 | - | - | - | - | 2 |
| CO5 | L2 | - | - | - | - | - | - | - | - | 3 | 3 | - | 2 |

## Textbooks

*(Suggested — not specified in the source.)*

- R. K. Bansal, Fluid Mechanics and Hydraulic Machines, Laxmi Publications.
- Modi & Seth, Hydraulics and Fluid Mechanics Including Hydraulic Machines, Standard Book House.

## Online Learning Resources

- NPTEL – Fluid Mechanics Laboratory Video Series – IIT Dharwad.
- Steve Brunton – Fluid Dynamics Data Analytics & Modeling Tutorials.
- OpenCV Image Feature Engineering Documentation & Scikit-Learn Regression Tutorials.

## Recommended Free & Open-Source Fluid Mechanics AI Tools

**Language & Environment:** Python, Scikit-Learn – regression modelling and data analytics on flow and turbine data.

**Vision & Signal Analysis:** OpenCV – image-based feature engineering for flow-visualization experiments.

**Machine Learning:** Deep Neural Network surrogates for turbine-efficiency mapping and cavitation anomaly detection.

II Year II Sem

# MANUFACTURING PROCESSES LAB

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AME3402P | II Year II Sem | 0 | 0 | 0 | 3 | 1.5 |

**Prerequisites:** Engineering Workshop Lab, Engineering Materials
**Branch:** Common to Mechanical and Allied Branches (Inferred – not specified in the source.) |
Regulation: JNTUA R26 (Inferred – assumed consistent with companion syllabus.)

## Course Objectives

- Introduce fundamental manufacturing processes including casting, forming, welding, and moulding; acquire practical knowledge of metal casting.
- Understand sand properties for metal casting.
- Familiarize different welding methods and practical skills in pattern making, sand testing, and joint fabrication.
- Gain knowledge of the working and processing of plastics.
- Develop simple parts on a 3D printing machine.

## List of Experiments

### MODULE: LIST OF EXPERIMENTS

- **1. Design and making of pattern —** (i) single piece pattern, (ii) split pattern.
- **2. Sand properties testing —** (i) sieve analysis (dry sand), (ii) clay content test, (iii) moisture content test, (iv) strength test (compression & shear), (v) permeability test.
- **3. Mould preparation —** (i) straight pipe, (ii) bent pipe, (iii) dumb-bell, (iv) gear blank.
- **4.** Gas cutting and welding.
- **5. Manual metal-arc welding —** (i) lap joint, (ii) butt joint.
- **6.** Injection moulding.
- **7.** Blow moulding.
- **8.** Simple models using sheet-metal operations.
- **9.** Study of deep-drawing and extrusion operations.
- **10.** Making weldments using TIG / MIG welding.
- **11.** Welding using a spot welding machine.
- **12.** Joining using brazing and soldering.
- **13.** Making simple parts on a 3D printing machine.
- **14.** Demonstration of metal casting.

## AI Tools Integration

### Casting

- **Casting Simulation Software:** MAGMA, ProCAST.
- **CAD/CAM:** Pattern design.
- **AI Defect Detection:** Computer vision for casting inspection.
- **ML Models:** Defect prediction.
- **Process Optimization:** AI-based parameter selection.
- **3D Printing:** Rapid pattern making.

### Welding

- **Welding Simulation Software:** Visual Environment, SYSWELD.
- **Computer Vision:** Weld defect detection.
- **AI Welding Robots:** Adaptive welding.
- **ML Models:** Weld quality prediction.
- **IoT:** Real-time weld monitoring.

- **AR:** Welding training systems.

### AI Application

- Machine Learning (ML) regression models to predict mould defects (e.g., blowholes, scabs) based on real-time sand testing inputs (moisture %, clay %, permeability number).
- Real-time weld defect detection (porosity, spatter, undercut) utilizing Convolutional Neural Networks (CNNs).
- Topographic optimization algorithms driven by AI to minimize component weight while maximizing strength; layer-by-layer optical print error correction.

## VIRTUAL LAB

- **1.** Study and observe various stages of casting through demonstration of the casting process. (virtual-labs.github.io/exp-sand-casting-process-dei)
- **2.** Weld and cut metals using an oxyacetylene welding setup. (virtual-labs.github.io/exp-gas-cutting-processes-iitkgp)
- **3.** Simulate the Fused Deposition Modelling (FDM) process. (3dpdei.vlabs.ac.in/exp/simulation-modelling-process)
- **4.** Mould-flow / topology simulation reference. (altair.com/inspire-mold)
- **5.** Cartesian-system simulation reference. (virtual-labs.github.io/exp-simulation-cartesian-system-dei)

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Make moulds for sand casting. | L3 (Apply) |
| CO2 | Fabricate different types of components using various manufacturing techniques. | L3 (Apply) |
| CO3 | Adapt unconventional manufacturing methods. | L3 (Apply) |
| CO4 | Develop different weld joints. | L3 (Apply) |
| CO5 | Explain different types of 3D printing techniques. | L2 (Understand) |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L3 | 3 | 2 | 2 | 2 | 1 | - | 1 | - | 2 | 1 | - | 2 |
| CO2 | L3 | 3 | 1 | 2 | 2 | 1 | 2 | - | - | 3 | 1 | - | 2 |
| CO3 | L3 | 3 | 2 | 1 | 2 | 2 | - | 1 | - | 2 | 1 | - | 1 |
| CO4 | L3 | 2 | 2 | 3 | 1 | 3 | - | - | - | 2 | 2 | 1 | 3 |
| CO5 | L2 | 2 | 3 | 3 | 3 | 3 | 1 | 1 | - | 2 | 2 | - | 3 |

## Textbooks

*(Suggested — not specified in the source.)*

- P. N. Rao, Manufacturing Technology: Foundry, Forming and Welding, McGraw Hill Education.
- Amitabha Ghosh & Asok Kumar Mallik, Manufacturing Science, Affiliated East-West Press.

## Online Learning Resources

- Virtual Labs – Sand Casting Process (DEI) – virtual-labs.github.io
- Virtual Labs – Gas Cutting Processes (IIT Kharagpur) – virtual-labs.github.io
- Virtual Labs – FDM Simulation (DEI) – 3dpdei.vlabs.ac.in

## Recommended Free & Open-Source Manufacturing AI Tools

**Casting & CAD/CAM:** MAGMA, ProCAST, computer-vision-based casting inspection, and ML-based defect prediction.

**Welding:** Visual Environment, SYSWELD, CNN-based weld defect detection, and IoT-based real-time weld monitoring.

> **3D Printing:** AI-driven topology optimization and layer-by-layer optical print error correction.

**II Year II Sem**

# EMBEDDED SYSTEMS AND IOT LAB

| Course Code | Year/Sem | L | T | Pr | P | Credits |
|---|---|---|---|---|---|---|
| 26AEC404S | II Year II Sem | 1 | 0 | 0 | 2 | 2 |

**Prerequisites:** Fundamental Python & C Programming
**Branch:** Common to CSE and Allied Branches (Inferred – not specified in the source.) | Regulation: JNTUA R26 (Inferred – assumed consistent with companion syllabus.)

## Course Objectives

- Learn to connect sensors, actuators, and communication modules with microcontrollers.
- Gain proficiency in Arduino IDE, Python IDE, and Raspbian OS environments.
- Implement serial, SPI, and wireless protocols like Zigbee for data transfer.
- Introduce the Raspberry Pi platform, which is widely used in IoT applications.
- Introduce the implementation of distance sensors on IoT devices.

## List of Experiments

### MODULE A: EMBEDDED SYSTEMS EXPERIMENTS

**Suggested Tools:** Arduino Duemilanove Board, Arduino Software IDE

**Experiments: (Any 5 of the following)**

- Measure analog signal from a temperature sensor.
- Generate PWM output.
- Drive single character generation on Hyper Terminal.
- Drive a given string on Hyper Terminal.
- Full duplex link establishment using Hyper Terminal.
- Drive a given value on an 8-bit DAC using SPI.
- Drive a stepper motor using analog GPIOs.
- Drive an accelerometer and display the readings on Hyper Terminal.

### MODULE B: INTERNET OF THINGS EXPERIMENTS

**Suggested Tools:** Raspberry Pi, Raspbian OS, Python IDE, Arduino, NodeMCU, Zigbee modules

**Experiments: (Any 5 of the following)**

- Getting started with Raspberry Pi; install Raspbian on the SD card.
- Set up a Python-based IDE for the Raspberry Pi and trace/debug Python code on the device.
- Using Raspberry Pi: (a) calculate distance using a distance sensor, (b) basic LED functionality.
- Interact with online services from the Raspberry Pi through public APIs and SDKs.
- Study and install the Arduino IDE and explore different types of Arduino boards.
- Study and implement the Zigbee protocol using Arduino / Raspberry Pi.
- Calculate distance using a distance sensor with Arduino.
- Basic LED functionality using Arduino and NodeMCU.
- Calculate soil moisture content using Arduino and NodeMCU.
- Calculate distance using a distance sensor with NodeMCU.
- Basic LED functionality using NodeMCU.

## Hands-on Practices

- Measure analog signal from a temperature sensor.
- Generate PWM output to control LED brightness / motor speed.
- Drive single character generation on Hyper Terminal.

- Drive a given string on Hyper Terminal.
- Full duplex link establishment using Hyper Terminal.
- Drive a given value on an 8-bit DAC using SPI.
- Drive a stepper motor using GPIOs.
- Interface an accelerometer and display readings on Hyper Terminal.

## AI Integration & Smart Applications

To modernise this lab, traditional sensor outputs can be linked to AI workflows:

- **Predictive Maintenance (Exp 1, 7, 8):** Feed temperature and accelerometer data into an Edge AI TinyML model to predict stepper motor failure.
- **Intelligent Gesture Control (Exp 8):** Use Raspberry Pi to run a lightweight Python classifier that maps accelerometer patterns to specific UI actions.
- **Smart Agriculture Analytics (Exp 10):** Send soil moisture data to a cloud-based AI system (e.g., OpenAI API or AWS IoT Analytics) to forecast irrigation needs based on weather data.

*Components / Boards: Arduino Duemilanove Board, Arduino Software IDE.*

## Course Outcomes (COs)

| CO | Course Outcome Statement | Bloom's Level |
|---|---|---|
| CO1 | Interface analog and digital sensors with Arduino, NodeMCU, and Raspberry Pi. | L3 (Apply) |
| CO2 | Implement wired (Serial, SPI) and wireless (Zigbee) communication protocols. | L3 (Apply) |
| CO3 | Develop Python and C/C++ scripts to process environmental data and control actuators. | L3 (Apply) |
| CO4 | Understand the concept of M2M (machine to machine) communication and its protocols, and implement distance sensors. | L2 (Understand) |
| CO5 | Realize the role of the internet in mobile devices, cloud computing, and sensor networks. | L2 (Understand) |

## CO – PO Articulation Matrix

Scale: 3 – High/Substantial, 2 – Medium/Moderate, 1 – Low/Slight, – No Correlation

| CO | Bloom's Level | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | L3 | 3 | 2 | 2 | - | 3 | - | 1 | - | 2 | 1 | - | 3 |
| CO2 | L3 | 3 | 3 | 2 | 1 | 3 | - | - | - | 2 | 1 | - | 3 |
| CO3 | L3 | 2 | 3 | 3 | 2 | 3 | - | - | - | 2 | 1 | - | 2 |
| CO4 | L2 | 3 | 3 | 3 | 2 | 3 | - | - | - | 2 | 1 | - | 2 |
| CO5 | L2 | - | - | - | - | - | 2 | 2 | 1 | 3 | 2 | 1 | 2 |

## Textbooks


- Raj Kamal, Embedded Systems: Architecture, Programming and Design, McGraw Hill Education.
- Arshdeep Bahga & Vijay Madisetti, Internet of Things: A Hands-On Approach, Universities Press.
- Pete Warden & Daniel Situnayake, TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, O'Reilly Media.
- Tammy Noergaard, Embedded Systems Architecture, Elsevier Publications, 2013.
- Shibu K.V, Embedded Systems, Tata McGraw Hill Education Private Limited, 2013.
- Frank Vahid & Tony Givargis, Embedded System Design, John Wiley Publications, 2013.
- Lyla B. Das, Embedded Systems, Pearson Publications, 2013.


## Reference Books


- Muhammad Ali Mazidi, Janice Gillispie Mazidi & Rolin D. McKinlay, The 8051 Microcontroller and Embedded Systems using Assembly and C, Pearson Education.
- Claire Rowland et al., Designing Connected Products: UX for the Consumer Internet of Things, O'Reilly Media.

## Online Learning Resources

- NPTEL – Introduction to Embedded Systems – IIT Kharagpur.
- Harvard EdX – TinyML Specialization Video Lectures.
- Edge Impulse – Official Firmware Engineering and Deployment Tutorials.

## Recommended Free & Open-Source Embedded / IoT AI Tools

**Language & Environment:** Arduino IDE, Python IDE, Raspbian OS – firmware development and device programming.

**Hardware Platforms:** Arduino, NodeMCU, Raspberry Pi – sensor interfacing and communication protocol experiments.

**Edge AI / TinyML:** TensorFlow Lite for Microcontrollers – predictive maintenance and gesture-classification models.

**Cloud & Analytics:** OpenAI API, AWS IoT Analytics – cloud-based forecasting of sensor data trends.

**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 YouTube