24MA101 **Linear Algebra, Series and Calculus**

Category: Basic Sciences (BS) **3L 0T 2P 4C**

Pre-requisite: 10+2 Mathematics

Course Description:

An overview of the fundamental concepts of linear algebra, infinite series, differential calculus, multiple integrals and vector calculus, with a focus on the applications in solving engineering problems

Course Aims and Objectives:

- Introduce techniques for solving systems of linear equations, determining eigenvalues and eigenvectors, and performing matrix diagonalization
- Explain methods to analyze the convergence and divergence of an infinite series and to expand functions using Taylor's or Maclaurin's series
- Familiarize differentiation rules and theorems to solve problems related to rates of change and optimization
- Explain the concept of double and triple integrals to calculate areas and volumes for two-dimensional and three-dimensional objects
- Teach operations on vector-valued functions, including line and surface integrals, as well as the concepts of curl and divergence, and their applications in engineering

Course Outcomes:

At the end of the course, the student will be able to…

- CO 1: solve the systems of equations and analyze engineering problems using linear algebra techniques [K3]
- CO 2: apply the convergence tests of an infinite series to solve engineering problems [K3]
- CO 3: use differential calculus to solve optimization problems and analyze rates of change in engineering applications [K3]
- CO 4: calculate areas and volumes using double and triple integrals[K3]
- CO 5: apply vector calculus concepts to solve problems involving work done by force fields and analyze related physical phenomena [K3]

Course Structure:

Unit 1: Linear Algebra

- **Contents**
  Rank of a matrix (Echelon form), Finding the inverse by Gauss-Jordan method, System of linear equations: Homogeneous and Non-Homogeneous, Linear transformations,

Orthogonal transformation, Eigenvalues and Eigenvectors, Reduction to Diagonal form

- **Description:** This unit familiarizes the students, the concept of linear algebra, which is essential for solving system of equations, analyzing data, modeling engineering problems
- **Exercises/Projects**:
  - Find the inverse of a matrix by Gauss-Jordan method
  - Solve a system of linear equations
  - Find the eigenvalues and eigenvectors of a given matrix
  - Diagonalization of a square matrix
- **Examples/Applications/Case Studies:**
  - Solve the system of linear equations representing currents in an electrical network to determine the voltage at each node
  - Diagonalize a symmetric matrix representing a physical system to simplify calculations of its behavior
  - Transforming a quadratic form into a canonical form for retrieval of shapes
- **Learning Outcomes:**
  - Find the rank of a matrix and use it to test for consistency of linear systems
  - Find eigenvalues and eigenvectors of a given matrix and perform diagonalization
- **Specific Resources:**
  - MIT OpenCourseWare: Linear Algebra https://ocw.mit.edu/courses/18-06-linear-algebra-spring-2010/
  - NPTEL: Linear Algebra https://archive.nptel.ac.in/courses/111/104/111104137/

Unit 2: Infinite Series

- **Contents**

  Infinite Sequence (Definition), Infinite Series, Comparison Tests, Integral Test, Ratio and Root Test, Alternating Series, Absolute and Conditional convergence
- **Description:** This unit covers infinite series, their convergence/ divergence, and applications in various engineering fields
- **Exercises/Projects:**
  - Finding the convergence/divergence of a given infinite series using appropriate tests
- **Examples/Applications/Case Studies:**
  - A ball is dropped from a height of 4m. Each time it strikes the pavement after falling from a height of h meters it rebounds to a height of 0.5h meters. Find the total distance the ball travels up and down
  - Define the sequence of partial sums $s_k = \sum_{n=1}^{n=k} \frac{1}{(sin^2 n)\, n^3}$. What happens, when

you try to find the limit of s_k as $k \to \infty$

- Apply the concept of series, to model periodic phenomena

- **Learning Outcomes:**
  - Apply various tests to determine the convergence/divergence of an infinite series
- **Specific Resources:**
  - Khan Academy: Infinite Series
    https://www.khanacademy.org/math/ap-calculus-bc/bc-series-new
  - NPTEL : Real Analysis I
    http://archive.nptel.ac.in/courses/111/106/111106142/

Unit 3: Differential Calculus

- **Contents**

  Mean value theorems: Rolle's theorem (without proof), Lagrange's mean value theorem (without proof), Taylor's and Maclaurin's theorems with Lagrange's form of remainder (without proof), Expansions of functions: Taylor's and Maclaurin's series

  Functions of Several Variables: Maxima and Minima of functions of two variables, Lagrange's method of undetermined multipliers

- **Description:** This unit focuses on expansion of functions as Taylor's and Maclaurin's series and finding extreme values of multi-variable functions with and without constraints
- **Exercises/Projects**:
  - Verification of mean value theorems
  - Taylor's and Maclaurin's series expansions of single variable functions
  - Examine the extreme values of a function
- **Examples/Applications/Case Studies:**
  - It took 14sec for a mercury thermometer to rise from -19^{0}C to 100^{0}C, when it was taken from a freezer and placed in boiling water. Show that somewhere along the way the mercury was rising at the rate of 8.5^{0}C per second.
  - On our moon, the acceleration of gravity is 1.6m/sec^2.If a rock is dropped into a crevasse, how fast will it be going just before it hits bottom 30sec later.
  - Show that the maximum value of $a^2b^2c^2$ on a sphere of radius r centered at the origin of Cartesian 'abc' coordinate system is $\left(\frac{r^2}{3}\right)^3$.
  - Show that for non-negative numbers a, b, c the geometric mean is less than or equal to their arithmetic mean.
- **Learning Outcomes**:
  - Apply the mean value theorems and find extreme values of multivariable functions

- **Specific Resources:**
  - MIT OpenCourseWare: Multivariable Calculus
    https://ocw.mit.edu/courses/18-02sc-multivariable-calculus-fall-2010/
  - NPTEL: Engineering Mathematics-I
    https://archive.nptel.ac.in/courses/111/105/111105121/

Unit 4: Multiple Integrals

- **Contents**

  Double integrals (Cartesian coordinates), Change of order of integration, Triple integrals, Change of variables to polar, cylindrical and spherical coordinates, Areas as double integration and Volumes as triple integration
- **Description:** This unit introduces double and triple integrals, which are essential for calculating areas and volumes of objects in two and three dimensions
- **Exercises/Projects:**
  - Evaluate a double integral to find the area enclosed by a region.
  - Evaluate a triple integral to find the volume of a solid
- **Examples/Applications/Case Studies:**
  - Calculating the center of mass using multiple integrals
- **Learning Outcomes:**
  - Evaluate double and triple integrals
  - Apply double and triple integrals to solve engineering problems involving areas and volumes
- **Specific Resources:**
  - MIT OpenCourseWare: Multivariable Calculus
    https://ocw.mit.edu/courses/18-02sc-multivariable-calculus-fall-2010/

Unit 5: Vector Calculus

- **Contents**

  Introduction to Gradient of a scalar field, Divergence and Curl of a vector field, Line integral, Green's theorem in the plane (without proof), Surface integrals, Stoke's theorem (without proof) and Gauss divergence theorem (without proof)
- **Description:** This unit covers vector-valued functions, curl and divergence, line integrals and surface integrals, which are used to analyze various physical phenomena in engineering
- **Exercises/Projects:**
  - Find gradient, divergence and curl of a point functions
  - Verification of Greens, Stokes and Gauss divergence theorems
- **Examples/Applications/Case Studies:**
  - Calculate work done by a force field using line integral.
  - Calculate total flux across the surface using surface integral

- **Learning Outcomes:**
  - Perform operations on vector-valued functions
  - Apply Green's theorem, Stoke's theorem, and the Divergence theorem to convert line integrals to area integrals and surface integrals to volume integrals for solving engineering problems
- **Specific Resources:**
  - MIT OpenCourseWare: Multivariable Calculus
    https://ocw.mit.edu/courses/18-02sc-multivariable-calculus-fall-2010/

Textbook(s) / Reference(s)

Text book:

1. Weir Maurice D., Hass Joel & Giodano Frank R. (2013). *Thomas' Calculus.* (11th Edition). Pearson Education,inc..
2. Grewal B. S. (2017). *Higher Engineering Mathematics.* (44th Edition). Khanna Publishers.

References:

1. Kreyszig Erwin. (2013). *Advanced Engineering Mathematics.*(9th Edition).Wiley Publishers.
2. Ramana B.V.(2007). *Higher Engineering Mathematics.*Tata Mc.Graw Hill

Mapping of Course Outcomes to Program Outcomes:

PO / CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	M	M	-	-	L	-	-	-	-	-	-	-
CO2	M	M	-	-	L	-	-	-	-	-	-	
CO3	M	M	-	-	L	-	-	-	-	-	-	-
CO4	M	M	-	-	L	-	-	-	-	-	-	-
CO5	M	M	-	-	L	-	-	-	-	-	-	-

Key:

- M: Moderate emphasis
- L: Low emphasis
- H: High emphasis
- '-': Not applicable

24MA101 Linear Algebra, Series and Calculus (LAB)

Category: Basic Sciences (BS)

Pre-requisite: 10+2 Mathematics

Course Description:

An overview of the functions of MathWorks Symbolic Math Tool Box to solve problems in Matrix algebra, differential calculus, multiple integrals, vector calculus and to test the nature of series.

Course Objectives:

The objective of the lab course is to demonstrate built-in functions to

- find inverse, rank, eigen values, eigen vectors of a matrix and solution of system of linear equations.
- analyze the convergence and divergence of infinite series.
- find ordinary and partial derivatives
- expand functions as Taylor's and Maclaurin's series.
- determine extreme values of multi-variable function with and without constraints.
- Calculate areas and volumes using double and triple integrals.
- find gradient, divergent and curl.

Course Outcomes:

At the end of the course, the student will be able to…

- CO 1: find rank, solution of systems of equations and eigen values & eigen vectors of matrix [K3]
- CO 2: examine the convergence/divergence of infinite series. [K3]
- CO 3: use differential calculus to solve optimization problems. [K3]
- CO 4: calculate areas and volumes using double and triple integrals [K3]
- CO 5: find gradient, divergent and curl. [K3]

List of experiments:

Unit 1: CO 1 [K3]

1. inv() function to find inverse of a matrix
2. rank() function to find rank of a matrix
3. rref() function to solve system of linear equations
4. eig() function to find eigen values and eigen vectors of a matrix

- **Specific Resources:**
  MathWorks Linear Algebra Documentation
  Linear Algebra - MATLAB & Simulink - MathWorks India

Unit 2: CO 2 [K3]

1. vpa() function to evaluate numerically each term of series
2. symsum() function to test the nature of the series

- **Specific Resources:**
  MathWork Symbolic Calculus Tool Box
  Calculus - MATLAB & Simulink - MathWorks India

Unit 3: CO 3 [K3]

1. diff() function to find ordinary and partial derivatives
2. taylor() function to expand functions as Taylor's and Maclaurin's series
3. fmincon() function to find minimum of a function with constraints

- **Specific Resources:**
  MathWork Symbolic Math Tool Box
  Symbolic Math Toolbox Documentation - MathWorks India
  MathWork Optimization Tool Box
  Optimization Toolbox Documentation - MathWorks India

Unit 4: CO 4 [K3]

1. int() function to find double and triple integrals

- **Specific Resources:**
  MathWork Symbolic Calculus Tool Box
  Calculus - MATLAB & Simulink - MathWorks India

Unit 5: CO 5 [K3]

1. gradient() function to find gradient of a scalar point function
2. divergence() function to find divergence of a vector point function
3. curl() function to find curl of a vector point function

- **Specific Resources:**
  MathWork Symbolic Calculus Tool Box
  Calculus - MATLAB & Simulink - MathWorks India

24PH101 APPLIED PHYSICS

(CSE, AI&DS, AI&ML and IT branches only)

***Category*: Basic Sciences (BS)** **3L 0T 0P 3C**

***Pre – requisite*: 10 + 2 Physics**

Course Description:

Applied Physics is designed to provide a comprehensive understanding of the fundamental principles of physics and their practical applications in engineering contexts. This course provides a key insight to the principles of quantum mechanics, quantum computing, lasers, optical fibers, semiconductors, and basic electronics. Through a combination of theoretical lectures and problem-solving sessions, students will develop the necessary skills to analyze and solve engineering problems.

Course Objectives:

1. Explain the basic concepts of quantum mechanics and formalism of Schrodinger's equations.
2. Describe the principles of quantum computing and its applications in the present scenario.
3. Elucidate the basic concepts of Lasers and Optical fibers and their diverse applications in Science and Technology.
4. Explain the basic concepts of semiconductors with respect to energy bands and their applications.
5. Discuss the basic concepts of electronics in the design and fabrication of diode and transistors.

Course Outcomes:

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

1. describe the basic concepts of quantum mechanics and its mathematical frame (K2)
2. summarize the basic principles of quantum computing and quantum algorithms (K2)
3. identify different types of Lasers and Optical fibers and their applications (K2)
4. apply the concepts of semiconductors towards the classification of materials based on energy bands (K3)
5. illustrate the formation of PN junction diode and transistor, and their applications (K3)

Course Contents:

Unit 1: Lasers and Fiber Optics (8 hours)

- **Lasers**: Introduction, characteristics of laser, basic principles of lasers (absorption, spontaneous emission, and stimulated emission), requirements of lasers, different types of lasers: solid-state lasers (Ruby), gas lasers, (He-Ne), applications of lasers.
- **Fiber Optics**: Introduction, fundamentals of optical fiber, propagation of light through optical fiber, types of optical fibers, numerical aperture, fractional refractive index change, fiber optics in communication and its advantages.

Applications:

- Presently lasers are being widely used for numerous medical applications such as cancer treatment, ophthalmology, dermatology, and cosmetic applications
- Optical fibers are used to transmit telecommunication in telephones, internet, and cable television.

Exercises:

- Calculate the numerical aperture of an optical fiber.
- Determine the wavelength of light by laser diffraction grating.

Learning Outcomes:

- Identify the basic elements in optical fiber, different modes, and configurations.
- Explain the significance of fiber optics communication system in daily life.

Specific Resources:

1. Introduction to LASER, NPTEL – M. R. Shenoy, Professor, IIT – Delhi, https://onlinecourses. nptel.ac.in/noc21_ph01/preview
2. Fiber Optics, NPTEL – Vipul Rastogi, Professor, IIT – Roorkee, https://onlinecourses.nptel.ac.in/ noc20_ph07/preview

Unit 2: Semiconductor Physics (8 hours)

- **Semiconductor Physics**: Introduction, formation of energy bands, classification of crystalline solids, fermi level in intrinsic semiconductors, fermi level in extrinsic semiconductors, large band gap semiconductors, drift and diffusion currents, Einstein's equation, Hall effect and its applications.

Applications:

- Semiconductors are crucial for optoelectronic devices like solar cells, light emitting diodes (LEDs), and Semiconducting lasers.

Exercises:

- Determine the carrier concentration in a semiconductor using Hall effect.

Learning Outcomes:

- Describe the various semiconductor devices and determine drift, diffusion current densities and carrier concentration.

Specific Resources:

1. Introduction to Semiconductor Devices, NPTEL – Naresh Kumar Emani, Professor, IIT – Hyderabad, https://archieve.nptel.ac.in/courses/108/106/10810610=81/.

Unit 3: Basic Electronics (8 hours)

- **Basic Electronics**: Introduction, PN-junction diode, current – voltage characteristics of P N-junction diode under forward bias and reverse bias, Zener diode, bipolar junction transistor, biasing of pnp and npn, characteristics of common base, common collector and common emitter configurations, Transistor as an amplifier**.**

Applications:

- P N Junction diode as a rectifier and Zener diode as a voltage regulator

Exercises:

- Compare and contrast the I – V characteristics of a PN junction diode and Zener diode.

Learning Outcomes:

- Understand the principles of rectification and its applications in converting AC to DC and their circuit designs.

Specific Resources:

1. Fundamentals of Semiconductor Devices, NPTEL – Digbijoy N. Nath, IISc – Bangalore, https://archieve.nptel.ac.in/courses/108/108/108108122/#.

Unit 4: Quantum Mechanics (8 hours)

- **Quantum Mechanics**: Dual nature of light, matter waves and Debroglie's hypothesis, G. P. Thomson experiment, Heisenberg's uncertainty principle and its application (non – existence of electron inside nucleus), Schrödinger's time independent wave equation, physical significance of wave function, particle in a one-dimensional box.

Applications:

- Schrodinger wave equation helps in modelling the electronic devices like diodes, transistors, and solar cells.

Exercises:

- Calculate the probability density and normalize a wave function within a finite interval.

Learning Outcomes:

- Understand wavefunctions and their interpretation in quantum mechanics.
- Apply Schrodinger's wave equation to solve quantum mechanical problems.

Specific Resources:

1. Quantum Mechanics, NPTEL – S. Lakshmi Bala, Professor, IIT – Madras, https://nptel.ac.in/ courses/ 115106066.

Unit 5: Quantum Computing (8 hours)

- **Quantum Computing**: Sustainability of quantum system for information processing – classical Bits and Qu – Bits – Bloch's sphere – quantum gates – multiple Qu-Bits – advantages of quantum computing over classical computation.

Applications:

- Quantum computing is useful in the development of novel algorithms, error correction techniques, and hardware advancements that helps in Machine Learning and AI.

Exercises:

- Determine the quantum numbers that governs the spatial orientation of an atomic orbital.

Learning Outcomes:

- Explain foundational principles of quantum mechanics and to describe the differences between classical and quantum computing models.

Specific Resources:

1. Introduction to Quantum Computing: Quantum Algorithms and Qiskit, NPTEL – Prabha Mandayam, Prof., IIT – Madras, https://archieve.nptel.ac.in/courses/106/106/10610 6232/

Textbook(s) / Reference(s):

Textbooks:

1. Avadhanulu, M. N. (2019). *A textbook of engineering physics* (11th ed.). S. Chand Publishing.
2. Halliday, D. Resnick, R., & Walker, J. (2020). *Fundamentals of physics* (10th ed.). John Wiley & Sons.

Reference Books:

1. Pandey, B. K., & Chaturvedi, S. (2021). *Engineering Physics* (1st ed.). Cengage Learning.
2. Sharma, S., & Sharma, J. (2018). *Engineering Physics* (1st ed.). Pearson Education India.
3. Srinivasan, M. R. (2009). *Physics for Engineers* (1st ed.). New Age International.
4. Vijay Kumar, K. (2011). *Engineering Physics* (1st ed.). S. Chand Publishing.
5. Mc Mahon, D. (2007). *Quantum Computing Explained* (1st ed.). John Wiley & Sons.
6. Nielsen, M. A., & Chuang, I. L. (2001). *Quantum computation and quantum information* (1st ed.). Cambridge University Press.

Mapping of Course Outcomes to Program Outcomes:

Contribution of Course Outcomes towards achievement of Program Outcomes (L – Low, M - Medium, H – High)												
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
CO1	**H**		**M**									
CO2	**H**											
CO3	**H**		**L**									
CO4	**H**											
CO5	**H**	**L**			**L**							

******

24EN101 **COMMUNICATIVE ENGLISH** **3L 0T 3P 4.5C**

Category **HS**

Pre-requisites: **Intermediate English**

Course Description:
Communicative English course covers Intercultural Communication, AI in higher education, and Natural Language Processing. It emphasizes effective listening, presentation techniques, innovation, creativity, advanced reading, and technical writing skills, addressing verbal and nonverbal communication fundamentals, processes, barriers, and strategies for confidence and fluency, preparing students for career readiness and leadership.

Course Objectives:
1) Explain the verbal and nonverbal processes of Communication.
2) Familiarize active listening skills and explain the traits of a good listener.
3) Introduce speaking skills for effective presentations.
4) Teach reading techniques to enhance employability skills.
5) Instruct writing techniques to enhance employability skills.

Course Outcomes:
Upon successful completion of the course, the student will be able to
CO 1: Relate and recall the verbal and nonverbal processes of Communication.
CO 2: Demonstrate active listening skills and acquire the traits of a good listener.
CO 3: Apply English speaking skills to deliver effective presentations.
CO 4: Appraise reading techniques like skimming and scanning to enhance employability skills.
CO 5: Assess writing techniques like outlining and clustering to enhance employability skills.

Course Structure

UNIT – I:

Introduction to Communication:

Basics of Communication, Process of Communication, Levels of Communication, Verbal and Nonverbal Communication, Channels of Communication, Barriers to Communication, ***Intercultural Communication.***

Description:

The unit covers the basics and process of intercultural communication, including intrapersonal, interpersonal, group, and mass communication. It explores verbal and nonverbal communication, channels like face-to-face, written, and addresses barriers such as language differences, cultural misunderstandings, and psychological barriers, offering strategies to overcome these challenges.

Examples:

1) Imagine you are a project manager leading a team of engineers working on a global project. Your tea consists of members from various cultural backgrounds, including India, Japan, and the United States. Despi having a common goal, you notice that cultural differences are affecting communication and collaboration with the team.

Describe the specific cultural differences that could impact communication within your team. How do these differences manifest in terms of communication styles, decision-making processes, and conflict resolution? Based on your understanding of intercultural communication, propose strategies to enhance communication and foster a more cohesive team environment. Support your recommendations with examples and relevant theories of intercultural communication.

Exercises:

1) Imagine you are working on a project with a team from different cultural backgrounds. How would you describe the concept of Intercultural Communication and its relevance in today's globalized world to your team members?

2) You are presenting a seminar on Language and Communication. How would you explain the origin and development of the term 'Communication' to your students, using real-life examples to illustrate the basic principles of communication?

3) You are participating in a group discussion on Effective Communication. How would you define the Process of Communication and elaborate on its different levels to the group?
4) Picture yourself in a scenario where you are training new employees. How would you differentiate Verbal communication from Nonverbal communication, emphasizing the importance of nonverbal cues in effective communication?
5) You are consulting a colleague about communication strategies in your institution. How would you describe the various Channels of Communication, using examples from your institution's current communication practices?
6) Imagine you are in a meeting discussing communication challenges. How would you identify and explain the common communication barriers that can disrupt effective communication processes, using relevant examples or scenarios to illustrate your points?

Learning Outcome: Relate and recall the verbal and nonverbal processes of Communication.

Specific Resources:

AI Tools:

Vosaic
https://vosaic.com/

Zoomi.ai.
https://zoomi.ai/

Culturewizard
https://www.rw-3.com/

UNIT – II:

Listening Skills:

Effective Listening, Active Traits of a Good Listener, Listening Modes, Types of Listening, Barriers to Effective Listening, Listening for general content, ***AI in Higher Education: Opportunities, Challenges, and Ethical Considerations.***

Description:

The unit explores AI's transformative role in higher education, emphasizing opportunities, challenges, and ethical considerations. It covers AI's impact on learning, personalization, and administrative efficiency. Additionally, it focuses on Effective Listening Skills, Modes, Barriers, and Strategies, helping students develop crucial academic and professional listening abilities for success.

Examples:

1) You are a student in an engineering program that has recently integrated AI technologies into its curriculu The use of AI is intended to enhance learning experiences and streamline administrative processes. Howeve some students and faculty members are concerned about the ethical implications of AI in higher education.

Describe the opportunities that AI presents in higher education, specifically within the context of your engineering program. What challenges do you foresee in implementing AI technologies, and how might these challenges be addressed? Additionally, discuss the ethical considerations surrounding the use of AI in education. Based on your analysis, propose guidelines or best practices for the responsible use of AI in higher education, taking into account the perspectives of students, faculty, and administrators.

Exercises:

1) You are a professor designing a new course curriculum. How would you incorporate AI to enhance personalized learning experiences for your students, and what benefits do you think this approach would offer to both students and educators?
2) Imagine you are mentoring a student who wants to improve him / her listening skills. How would you define listening and explain the characteristics of a good listener to him / her?
3) You are facilitating a workshop on effective communication. How would you describe the various modes of listening, and how do they impact communication?
4) Picture yourself in a group discussion on different types of listening. How would you explain each type of listening, providing examples?
5) You are leading a training session on effective listening. How would you analyse the barriers to

effective listening and suggest strategies to overcome each barrier?

6) You are preparing a study guide for students. How would you explain the relationship between listening and taking notes, and why is this correlation important for academic success?

Learning Outcome: Demonstrate active listening skills and acquire the traits of a good listener.

Specific Resources:

AI Tools:

Elsa Speak
https://elsaspeak.com/en/

Listenwise
https://listenwise.com/

Duolingo
https://www.duolingo.com/

UNIT – III:

Speaking Skills:

Achieving Confidence, Clarity & Fluency, Effective Presentation Strategies, Meetings, Conferences, Paralinguistic Features, Barriers to Speaking, ***Natural Language Processing.***

Description:

The unit explores advanced communication skills and Natural Language Processing (NLP) fundamentals, enhancing Confidence, Clarity, and Fluency. It covers Effective Presentation Strategies, Meeting and Conference participation, and Paralinguistic features like tone and body language. Additionally, it identifies speaking barriers and provides techniques for clear, and impactful professional communication.

Examples:

1) You are a member of an engineering team tasked with presenting a proposal for a new project to a group stakeholders. During the presentation, you notice that your team members' paralinguistic features, such as tone voice, pitch, and pace of speech, significantly impact the audience's perception of the proposal. Some tea members speak too quickly, making it difficult for the audience to follow, while others use a monotone voic which affects the overall engagement.

Identify and analyse the paralinguistic features displayed by your team members during the presentation. How do these features influence the audience's perception of the proposal? Based on your analysis, propose strategies to improve paralinguistic communication within your team, ensuring that future presentations are engaging and effective. Support your recommendations with examples and relevant theories of paralinguistic communication.

Exercises:

1) Imagine you are explaining to a friend how your voice assistant understands and responds to your commands. Can you describe how Natural Language Processing (NLP) facilitates this interaction and its main goals?
2) Picture yourself preparing for a public speaking event. How do you think focusing on confidence, clarity, and fluency will impact your effectiveness as a speaker? Share a personal experience or a hypothetical scenario.
3) You have been asked to give a presentation to a group of potential clients at your workplace. Why is making presentations important in the corporate sector, and what strategies will you use to ensure that your presentation is effective?
4) Suppose you are tasked with organizing a major conference at your company. What are the crucial points you need to remember to ensure the event is successful?
5) Think about a time when you have to speak clearly and choose your words carefully. How does the Oliver Wendell Holmes Sr. quote "Speak clearly, if you speak at all; care every word before you let it fall" relate to the importance of paralinguistic features in effective speaking?
6) Imagine you are mentoring a peer who struggles with public speaking. What barriers to speaking might they face, and what advice would you give them to overcome these challenges? Share specific strategies or examples.

Learning Outcome: Apply English speaking skills to deliver effective presentations.

Specific Resources:

AI Tools:

Speechace
https://www.speechace.com/

Rosetta Stone
https://www.rosettastone.com/

ELSA Speak
https://elsaspeak.com/en/

UNIT – IV:

Reading Skills:

Reading Techniques, Techniques for Good Comprehension, Study Skills, Reading and Interpretation, Intensive, Extensive & Critical Reading, Reading for Different Purposes, ***Innovation and Human Creativity.***

Description:

The unit explores the synergy between Innovation, Creativity, and Advanced Reading Skills. It covers essential reading techniques for Comprehension, including Intensive, Extensive, and Critical Reading methods. Students will learn to adapt strategies for detailed understanding, broad knowledge, or critical analysis, fostering academic excellence and innovative thinking.

Examples:

1) As an engineering student, you are part of a team working on a project to develop a new renewable ener technology. Despite having access to the latest research and resources, the team is struggling to come up wi innovative solutions. The project deadline is approaching, and there is pressure to deliver a ground-breaki solution.

Identify and analyse the factors hindering innovation and creativity within your team. How do these factors impact the team's ability to generate innovative ideas? Based on your analysis, propose strategies to foster a culture of innovation and creativity within your team. Support your recommendations with examples and relevant theories of innovation and creativity.

Exercises:

1) Imagine you are part of a team working on a project to address a global challenge, such as climate change. How would you evaluate the importance of innovation and human creativity in finding solutions to this challenge?
2) Picture yourself preparing for a difficult exam. How would you explain the relevance of the SQ3R technique and describe how it helps you understand complex passages?
3) You are tutoring a student who is struggling academically. How would you explain the importance of study skills to him / her and how these skills can impact his / her academic performance?
4) Suppose you are leading a workshop on communication skills. How would you outline the importance of reading techniques in improving communication skills to the participants?
5) Imagine you are in a book club discussing a novel. How would you discuss the need to understand the author's point of view when reading and interpreting the text with the other members?
6) You are mentoring a group of students on effective reading strategies. How would you sketch the significance of intensive, extensive, and critical reading in helping them understand a text better?
7) Picture yourself in a classroom discussing the quote by Francis Bacon "Reading maketh a full man; conference a ready man; and writing an exact man.". How would you explain the different purposes of reading and interpreting a text using this quote as a reference?

Learning Outcome: Appraise reading techniques like skimming and scanning to enhance employability skills.

Specific Resources:

AI Tools:

Readlang
https://readlang.com/

Spreeder
https://www.spreeder.com/

Voice Dream Reader –

https://www.voicedream.com/reader/

UNIT – V:

Writing Skills:

Technical Writing, Importance and Characteristics, Techniques for Good Technical Writing, Paragraph Construction, Essays, Reports, ***Communication for Career-readiness and Leadership.***

Description:

The unit develops essential Communication Skills for Career Readiness and Leadership, focusing on Technical Writing. Students learn techniques for creating effective Reports, Essays, and Emails, emphasizing Clarity, Coherence, and Conciseness. It covers Paragraph Construction and organizing technical information, equipping students with effective professional communication and enhanced career prospects.

Examples:

1) You are a recent engineering graduate who has been hired by a leading technology company. As part of yo role, you are required to communicate technical information to non-technical stakeholders, including clients ai senior management. However, you are struggling to adapt your communication style to suit different audiences.

Describe the challenges you face in communicating technical information to non-technical stakeholders. How do these challenges impact your ability to advance in your career and assume leadership roles? Based on your analysis, propose strategies to improve your communication skills and effectively convey technical information to diverse audiences. Support your recommendations with examples and relevant theories of communication for career readiness and leadership.

Exercises:

1) Imagine you are mentoring a group of new employees. How would you explain how professional communication contributes to career readiness and effective leadership to them?
2) Picture yourself in a meeting with industry experts. How would you elaborate on why Technical Writing has become so important for industry and organizations, focusing on its importance and characteristic features?
3) You are preparing a technical manual for a new product. How would you ensure conciseness and flow in your Technical Writing?
4) Suppose you are teaching a class on writing skills. How would you explain the important points to remember while constructing an effective paragraph to your students?
5) You are writing an article for a leadership magazine. How would you compose a paragraph on the "Importance of Leadership Skills"?
6) Imagine you are preparing to write an essay. How would you illustrate the steps involved in developing a good essay, and how would you approach writing an argumentative essay on "Is Artificial Intelligence good for society?"
7) You are a journalist reporting on a science symposium. How would you discuss the structure and characteristics of a Formal Report, and how would you prepare a report on the science symposium event for your college magazine?

Learning Outcome: Assess writing techniques like outlining and clustering to enhance employability skills.

Specific Resources:

AI Tools:

ProWritingAid
https://prowritingaid.com/

Grammarly
https://languagetool.org/

eAngel
https://www.scribens.com/

Text Book:

1) Raman, M., & Sharma, S. (2012). *Technical Communication* (2nd ed.). Oxford University Press.

References:

1) Pushplata, & Kumar, S. (2011). *Communication Skills.* Oxford University Press.

2) Rizvi, A. (2005). Eff*ective Technical Communication*. Tata McGraw-Hill Publishing Company Limited.

3) Mishra, S, & Muralikrishna, C. (2006). *Communication Skills for Engineers*. Pearson.

Web Resources:

http://ndl.iitkgp.ac.in/he_document/ekumbh/ekumbh/85?e=5|text%20books%20on%20technical%20communication||| Kulbhushan Kumar. *English for Technical Professionals.*

https://www.gtuelibrary.edu.in/publication/Technical%20communication%205th%20June'09.pdf
Prof. (Dr.) M. D. Desai, *Technical Communication.*

https://gnindia.dronacharya.info/CSE-AI-ML/Common-Subjects/Downloads/Technical-Communication/Books/Technical-Communication-Book-1.pdf
Technical Communication Principles and Practice
Meenakshi Raman and Sangeeta Sharma

Contribution of Course Outcomes towards achievement of Program Outcomes (L – Low, M – Medium, H – High)												
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	M									H		M
CO2	M									H		M
CO3	M									H		M
CO4	M									H		M
CO5	M									H		M

HEAD
Department of English
V.R. Siddhartha Engg College
VIJAYAWADA - 7.

COURSE CODE: 24CS102
PROGRAMMING USING C

Course Category:	Engineering Science (ES)	**Credits:**	3
Course Type:	Theory	**Lecture -Tutorial-Practice:**	3-0-0
Pre-requisites:		**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	40 60 100

Course Description

This course introduces foundational programming concepts, covering algorithms, flowcharts, and pseudo code. It explores the C language structure, data types, operators, control structures, loops, arrays, strings, and functions. Advanced topics include pointers, dynamic memory allocation, structures, unions, enumerations, and file handling. Students gain hands-on experience with inter-function communication, recursion, sorting/searching techniques, and memory management. The course emphasizes practical coding skills, problem-solving, and program design, providing a solid foundation for software development using the C language.

Course Objectives

- *To introduce students to the fundamentals of computer programming.*
- *To provide hands-on experience with coding and debugging.*
- *To foster logical thinking and problem-solving skills using programming.*
- *To familiarize students with programming concepts such as data types, control structures, functions, and arrays.*
- *To encourage collaborative learning and teamwork in coding projects.*
- *To evaluate and apply C programming techniques proficiently for searching and sorting.*

Course Outcomes

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

	Course Outcomes	BTL	POI
CO1	Understand fundamental programming concepts in C through algorithms, flowcharts, and selection statements.	K2	1.7.1,2.5.1, 2.5.2, 2.7.1
CO2	Develop efficient C programs using loops, arrays, and strings using control structures.	K3	1.7.1,2.5.1, 2.5.2, 2.6.3,3.5.1
CO3	Implement modular C programs using functions, pointers, and memory optimizations.	K3	1.7.1, 2.5.2, 2.6.3, 3.5.1, 5.4.1
CO4	Develop C programs using structures and unions for user defined data types.	K3	2.5.2, 2.6.3,3.5.1
CO5	Analyze the use of enumerations and file handling techniques in C to manage data efficiently and solve real-world programming problems.	K4	1.7.1,2.5.1, 2.5.2, 2.6.3,3.5.1

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	2												
CO2	3	3	2										2	2
CO3	2	2	2		2								2	2
CO4	2	2	2										2	2
CO5	2	2	2										2	2

(1- Low, 2 - Medium, 3 – High)

Unit-I : Introduction to C & Problem solving

Introduction to the C Language: Introduction to Programming Languages, Basics of a Computer Program, Algorithms, flowcharts (Using Dia Tool), pseudo code. Introduction to Compilation and Execution, Background of C program, Identifiers, Types, Variables, Constants, Memory Layout, Input/Output, Programming Examples.

Structure of a C Program: Logical Data and Operators, Expressions, Precedence and Associatively, Evaluating Expressions, Type Conversion, Statements, Storage Class.

Selection: Two-way Selection, Multiway selection, More Standard Functions.

Unit-II: Repetition, Arrays & Strings

Repetition: Concept of Loops in C, Loop Examples, the Calculator Program.

Arrays: Array Concepts in C, Inter-Function Communication, Array Applications, One Dimensional Arrays, Linear Search and Binary Search Techniques, Selection Sort, Bubble Sort, Two Dimensional Arrays, Multidimensional Arrays.

Strings: String Concepts, C Strings, String Input/output Functions, Arrays of Strings, String Manipulation Functions, String- Data Conversion.

Unit-III: Functions, Pointers & Memory Allocations

Functions: Functions in C, User Defined Functions, Call by Value, Call Value Reference, Inter-Function Communication, Standard Functions, Scope, Recursion and advantages.

Pointers: Introduction to Pointers, Pointers for Inter-Function Communications, Pointers to Pointers, Compatibility, Lvalue and Rvlaue. Arrays and Pointers, Pointers Arithmetic and Arrays, Passing an Array to a Function, Array of Pointers.

Memory Allocation: Need of dynamic memory allocation, malloc(), calloc(), free(), realloc(), NULL, Stack vs. Heap Allocation.

Unit-IV: Structures & Unions

Structures: Structure Type Declaration, Initialization, Accessing Structures, Operations on Structures, Complex Structures, Structures and Functions, Sending the Whole Structure, Passing individual structure members, passing structure via pointer, nested structure.

Unions: Referencing Unions, Initializers, Unions and Structures, Internet Address, Programming Applications.

Unit-V: Enumerations & Files

Enumerations: The Type Definition (Typedef), Enumerated Types: Declaring an Enumerated Type, Operations on Enumerated Types, Enumeration Type Conversion, Initializing Enumerated Constants, Anonymous Enumeration: Constants, Input/Output Operators.

Files: introduction to the files, Uses of Files, Text files Vs. Binary files, Opening and closing FILE , Modes of FILE operation, Command line arguments, Standard Library Input /Output functions ,Character i/o functions, File Handling functions.

Text Books:

1. "The C Programming Language", Brian W. Kernighan and Dennis M. Ritchie, Prentice-Hall, 1988
2. Behrouz A. Forouzan and Richard F. Gilberg, "Computer Science A Structured Programming Approach Using C", CENGAGE Learning, Third Edition.
3. Schaum's Outline of Programming with C, Byron S Gottfried, McGraw-Hill Education, 1996

Reference Books:

1. Computing fundamentals and C Programming, Balagurusamy, E., McGraw-Hill Education, 2008.
2. Programming in C, Rema Theraja, Oxford, 2016, 2nd edition
3. C Programming, A Problem Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE, 3rd edition
4. Horowitz Sahni and Anderson-Freed, "Fundamentals of Data Structures in C", 2nd edition, Universities Press, 2011

Web Resources:

1. NPTEL, "NOC: Introduction to Programming in C," [Online]. Available: https://nptel.ac.in/courses/106/104/106104128/. [Accessed: Feb. 25, 2025].
2. Coursera, "C for Everyone: Structured Programming," [Online]. Available: https://www.coursera.org/learn/c-structured-programming/. [Accessed: Feb. 25, 2025].
3. GeeksforGeeks, "Arrays, Pointers and Functions in C," [Online]. Available: https://www.geeksforgeeks.org/arrays-and-pointers-functions-in-c/. [Accessed: Feb. 25, 2025].
4. GeeksforGeeks, "C Enumerations," [Online]. Available: https://www.geeksforgeeks.org/enumeration-enum-c/. [Accessed: Feb. 25, 2025].

5. Codeforwin, "Understanding malloc, calloc, realloc, and free," [Online]. Available: https://codeforwin.org/2018/07/malloc-calloc-realloc-free-functions-c-programming.html. [Accessed: Feb. 25, 2025].
6. GeeksforGeeks, "Unions in C," [Online]. Available: https://www.geeksforgeeks.org/union-c/. [Accessed: Feb. 25, 2025].
7. Programiz, "Structures in C," [Online]. Available: https://www.programiz.com/c-programming/c-structures. [Accessed: Feb. 25, 2025].
8. TutorialsPoint, "Formatting Input/Output Functions and Character Input/Output Functions," [Online]. Available: https://www.tutorialspoint.com/cprogramming/c_file_io.htm. [Accessed: Feb. 25, 2025].
9. Programiz, "Command-Line Arguments," [Online]. Available: https://www.programiz.com/c-programming/c-command-line-arguments. [Accessed: Feb. 25, 2025].
10. W3Schools, "C Programming Language," [Online]. Available: https://www.w3schools.com/c/index.php. [Accessed: Feb. 25, 2025].

COURSE CODE: 24CS181
PROGRAMMING USING 'C' LAB

Course Category:	Engineering Science (ES)	**Credits:**	1.5
Course Type:	Practical	**Lecture -Tutorial-Practice:**	0-0-3
Pre-requisites:		**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	60 40 100

Course Description

This hands-on course introduces students to the foundational principles of the C programming language, integrating essential Linux command-line skills and compiler tools such as GCC and Turbo C. The course emphasizes problem-solving, logical thinking, and algorithm development. Students will work through a structured progression from simple input/output to complex file operations and data structures, including practical projects and assignments.

Course Objectives

The primary objective of this C Programming Lab course is to provide hands-on practical experience in writing, debugging, and executing C programs. It aims to develop students' understanding of fundamental programming concepts such as input/output operations, control structures, arrays, strings, functions, and pointers through practical exercises. By the end of the course, students will be proficient in using essential C programming tools, developing logical problem-solving skills, and applying programming concepts to build efficient programs in a Linux environment using compilers like Turbo C and GCC.

Course Outcomes

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

	Course Outcomes	BTL	POI
CO1	Understand basic Linux shell commands to compile and run C programs using GCC.	K2	1.7.1,2.5.1, 2.5.2, 2.7.1
CO2	Select the right control structure for solving the problem.	K3	1.7.1, 2.5.1,2.5.2, 2.6.3,3.5.1
CO3	Develop C programs which utilize memory efficiently using programming constructs like pointers.	K3	1.7.1, 2.5.2, 2.6.3,3.5.1 5.4.1
CO4	Develop, Debug and Execute programs to demonstrate the applications of arrays, functions, basic concepts of pointers in C.	K3	2.5.2, 2.6.3,3.5.1
CO5	Analyze the use of enumerations and file handling techniques in C to manage data efficiently and solve real-world programming problems.	K4	1.7.1, 2.5.1,2.5.2, 2.6.3,3.5.1

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2	2												
CO2	3	2	3										2	2
CO3	3	2	3		2								2	2
CO4	3	2	3										2	2
CO5	2	3	3										2	2

(1- Low, 2 - Medium, 3 – High)

Course Content:

WEEK- 1

1. Basic Linux environment and its editors like Vi, Vim & Emacs etc.

2. Exposure to Turbo C, gcc

3. Write a C program to print the following output Input:

Enter a Character: *
Enter a Number: 1
Output:

```
* * * * * * *
*       1     *
*       1     *
*       1     *
*       1     *
*     1 1 1   *
* * * * * * *
```

4. Write a C program that takes two numbers and an arithmetic operator (+, -, *, or /) as input from the user using scanf(), performs the corresponding operation, and displays the result using printf().

5. Write a C program that reads two numbers from the user using `scanf()`, and then swaps their values using both methods:

1. With a third variable
2. Without using a third variable

Finally, display the swapped values using `printf()`.

6. Declare three variables of integer data type. User has to input three valid numbers. Display Sum and average of user entered numbers.
Approach is to take three numbers and find their sum and average using the formula given below-

Sum: a+b+c
Average: (a+b+c)/3
Where a,b,c are the three numbers.
Sample Input:
Enter 3 Values: 10 20 30
Sample Output: Sum: 60 Average: 20

WEEK-2

1. Input the temperature in Fahrenheit and output the equivalent temperature in Celsius and Vice – Versa. Input two numbers. The first is for a Celsius value, and the second is for a Fahrenheit value. Input Celsius value, convert it to Fahrenheit. Use the formula F = C*9/5 + 32 for conversion. In case of decimals, show up to 1 decimal value.
Input the Fahrenheit value, convert it to Celsius.
Use the formula C= (F−32)×5/9 for conversion. In case of decimals, show up to 1 decimal value.
Sample Input: 0 100
Sample Output: 32 37.7

2. You can calculate a Simple Interest by just providing the Principle Amount, Rate of Interest and Time or Periods provided by the user input. We can calculate the Simple Interest by the using the below Formula.

Simple Interest = (Principal*Rate*Time)/100

Principal (P): The principal is the amount that was initially borrowed (loan) from the bank or invested.

Rate (R): It is the rate of interest at which the principal amount is given to someone for a certain time; the rate of interest can be 5%, 10%, or 13%, etc.

Time (T): Time is the duration for which the principal amount is given to someone.

Constraints: 1<=Principal<=10000 1<=Rate<=10

1<=T<=30

Sample Input: 100 3 10

Sample Output: 30

3. You are working on a program that tracks the daily sales of a small bookstore. The owner wants to calculate the total sales and average sales for three consecutive days to better understand business trends. Write a program that takes the sales figures for three days as input, calculates the total sales, and finds the average sales. How will you implement this in C?

4. Write a C program to evaluate and display the result of the following expressions. Use the given variable declarations and initialize them with appropriate values. Also, print the results of each expression.

- A+B*C+(D*E) + F*G
- A/B*C-B+A*D/3
- A+++B---A
- J= (i++) + (++i)

5. Write a C program to calculate electricity bill according to the given condition:

For first 50 units Rs. 0.50/unit

For next 100 units Rs. 0.75/unit

For next 200 units Rs. 1.20/unit

For unit above 250 Rs. 1.50/unit

An additional surcharge of 20% is added to the bill.

WEEK- 3

1. To find factorial of the any given positive number. The factorial of a positive number n is given by: 1*2*3*4 ...Note: This program should take a positive integer from the user as the factorial of a negative number doesn't exist and, the factorial of 0 is 1. Compute the factorial using any loop. Since the factorial of a number
may be very large, the type of factorial variable is declared as unsigned long. If the user enters a negative number, the program should display a custom error message.

2. Input a number, check the given number is a prime or not. A prime number should be a natural number greater
than 1 that has no positive divisors other than 1 and itself.
Test Data and Output: Enter n: 5
Output: Prime
Enter n: 6
Output: Not Prime

3. Checking a number palindrome. Number should be a positive integer having more than one digit as all the single digits are palindromes.
Test Data and output:
Input : 2002 Input: 1234
Output: true Output: false

4. Construct a pyramid of numbers. A pyramid of numbers represents the number of individuals per unit area of various trophic areas where producers are kept at the base and the tip is occupied by top carnivores.

The pyramid of numbers is mostly upright. The members of successive higher trophic levels are higher than the previous one.

1. A higher trophic level has fewer individuals than that of the lower trophic levels.

1

2 2 2

3 3 3 3 3

4 4 4 4 4 4 4

5 5 5 5 5 5 5 5 5

5.Develop a C program that takes an integer as input from the user and prints 'Yes' if the number is an Armstrong number, and 'No' otherwise.(This adds specific input/output requirements.)

WEEK- 4

1. Write a C program to delete all duplicate elements from an array.
The program should:

- Ask the user to enter the number of elements in the array.
- Accept the array elements from the user.
- Remove all duplicate values from the array.
- Display the new array with only unique elements.

2. Write a C program to insert an element into a specific position in an array.
The program should:

- Ask the user to enter the size of the array and its elements.
- Ask the user to enter the element to insert and the position at which it should be inserted.
- Insert the element at the specified position by shifting the existing elements.
- Display the updated array.

3. Find 2's complement of the given binary number.
To get 2's complement of a binary number, simply invert the given number and add 1 to the least significant bit (LSB) of given result.
Test Data and Output:
Find 2's complement of binary number 10101110.
Simply invert each bit of given binary number, which will be 01010001. Then add 1 to the LSB of this result, i.e., 01010001+1=01010010 which is answer.
Find 2's complement of binary number 10001.001.
Simply invert each bit of given binary number, which will be 01110.110 Then add 1 to the LSB of this result, i.e., 01110.110+1=01110.111 which is answer.

4. Write a C program to sort array elements in ascending or descending order.

5. Given an array of integers **nums** and an integer **target,** return indices of the two numbers such that they add up to target. You may assume that each input would have exactly one solution and you may not use the same element twice. You can return the answer in any order.
Sample Input:
Enter the size of an array: 4
Enter array elements: 2 7 11 15
Enter target: 9
Output: [0,1]

WEEK- 5

1. A matrix can only be added to another matrix if the two matrices have the same dimensions. To add two matrices, just add the corresponding entries, and place this sum in the corresponding position in the matrix which results.
Input elements in 3x3 matrix1: 1 2 3
4 5 6
7 8 9
Input elements in 3x3 matrix2: 9 8 7
6 5 4
3 2 1
Sum of both matrix = 10 10 10
10 10 10
10 10 10

2. Matrix multiplication is a binary operation that produces a matrix from two matrices. For matrix multiplication, the number of columns in the first matrix must be equal to the number of rows in the second matrix. The resulting matrix, known as the **matrix product**, has the number of rows of the first and the number of columns of the second matrix. The product of matrices **A** and **B** is denoted as AB.

3. Concatenate two strings without using built-in functions
Note: User would be asked to enter two strings and then the program would concatenate them. For concatenation we have not used the standard library function strcat(), instead we have written a logic to append the second string at the end of first string.
Test Data and Output:
Str1: Good Str2: Morning Output: Good Morning

4. Reverse a string using built-in and without built-in string functions
Using built-in function: The function is used for reversing a string. The reversed string will be stored in the same string.

5. Input two strings **str1** in lowercase, **str2** in uppercase. Print the lower case string **str1** in uppercase and the uppercase string **str2** in lowercase.

WEEK- 6

1. Write a C program using a user-defined function to find the biggest number given any three numbers.

2. Write a **C program** to calculate the **factorial of a given number** using recursion. The program should prompt the user to enter a number and then compute its factorial using a recursive function.

3. Write a **C program** to generate the **Fibonacci series up to N terms** using a recursive function. The program should prompt the user to enter a number **N**, then compute and display the first **N Fibonacci numbers**.

4. Write a **C program** to swap two numbers using **call by value and call by reference**. The program should demonstrate that changes made inside the function do not affect the original values in the main function.

5. How can recursion be used to calculate NCR? Write a C program to demonstrate this.

WEEK- 7

1. String is a sequence of characters. Input two strings str1 and str2. Copy the contents of str1 to str2 using functions and pointers. Define a function copystr() with two pointer arguments copystr(*str1,*str2) Approach :
1. Scan string str from 0 to length-1.
2. check one character at a time based on ASCII values
- if(str[i] >= 65 and str[i] <=90), then it is uppercase letter, .
- if(str[i] >= 97 and str[i] <=122), then it is lowercase letter,
- if(str[i] >= 48 and str[i] <=57), then it is number,
- else it is a special character

3. Print all the counters
Sample Input:
Enter a string: Programming
Sample Output:
String1: Programming
String2: Programming

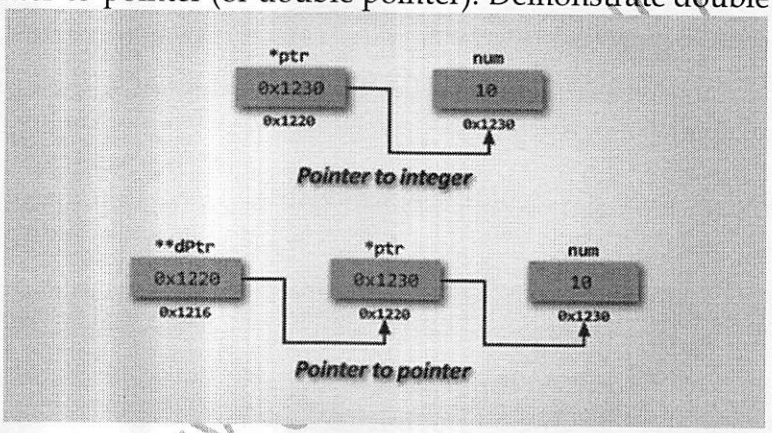
2. Given a string, write a program to count the occurrence of Lowercase characters, Uppercase characters, Special characters, and Numeric values. Define a function to count.
Input : #CseAi01dOr@gAIml07
Output :
Upper case letters : 5
Lower case letters : 8
Numbers : 4

Special Characters : 2
Input : *AiMlC4AiDsS*
Output :
Upper case letters : 6
Lower case letters : 4
Numbers : 1
Special Characters : 2

3. Write a c program to perform simple pointer arithmetic operations in c.

4. Write a c program to find sum of array elements by using array of pointers in c

5. When a pointer holds the address of another pointer then such type of pointer is known as pointer-to-pointer or double pointer. ptr is a normal pointer that holds the address of an integer variable num. There is another pointer **dptr in the diagram that holds the address of another pointer ptr, the pointer dptr here is a pointer-to-pointer (or double pointer). Demonstrate double pointers using the following representation.



WEEK- 8

1. Write a C program to define a structure named Student that contains the following members:

- name (character array)
- rollNumber (integer)
- age (integer)

Using a structures concept, store and display the details of two students.

.

2. Write a C program to define a structure named Student with the following members:

- name (character array) - to store the student's name.
- rollNumber (integer) - to store the roll number.
- marks (array of 3 integers) - to store marks obtained in 3 subjects.

Using an array of structures, read the details of two students from the keyboard. Calculate and display:

- Name of the student
- Roll number

- Marks in 3 subjects
- Total marks obtained

3. Write a C program to define a structure named Book with the following members:
- title (character array) – to store the book's title.
- author (character array) – to store the author's name.
- price (float) – to store the price of the book.

Using an array of structures, read the details of **five books** from the keyboard and display them in a formatted output.

4. Write a C program to define a structure **Student** that includes the following details:
- Name (character array)
- Roll Number (integer)
- Age (integer)
- Date of Birth (another structure with day, month, and year as integers)

Note: Student Details (with Date of Birth as a Nested Structure)

5. Write a C program to define a structure **Employee**, which contains the following details:
- **Employee Name** (character array)
- **Employee ID** (integer)
- **Salary** (float)

Additionally, the structure **Employee** should have a **nested structure Address**, which includes:
- **City** (character array)
- **State** (character array)
- **Pin Code** (integer)

The program should:
- Accept details of an employee, including their address.
- Display the entered details in a proper format.

Write the complete C program, including structure definition, input, and output statements.
Note: Employee Records (with Address as a Nested Structure)

WEEK- 9

1. Write a C program to demonstrate **passing individual structure members** to a function. Define a structure for a student with members: name, roll number, and marks. Pass individual members to a function to display the details.

2. Explain the concept of **passing an entire structure** to a function in C. Write a program to define a structure for an Employee with members: name, ID, and salary. Pass the entire structure to a function to display the details.

3. Define a union Data with members: i (integer), f (float), str (string). Write a program to assign and print values to each member of the union one by one. Observe what happens to the values.

4. Explain the concept of **passing a structure through pointers** in C. Write a program to define a structure for a Student with members: name, roll number, and marks. Pass the address of the structure to a function and display the student details using pointers.

5. Write a C program to **add two complex numbers**.

- Define a **structure** Complex with two members: real (to store the real part)
- imag (to store the imaginary part)
- Take input for two complex numbers from the user.

Perform the addition of the two complex numbers: Add the **real parts** separately.

- Add the **imaginary parts** separately.
- Display the **sum** of the two complex numbers in the form **(a + bi)**.

Sample output:
Enter first complex number: 3 2
Enter second complex number: 1 7
Sum: 4 + 9i

WEEK- 10

1. Write a C program that defines an enumeration for different car brands and asks the user to enter a number. Display the corresponding car brand.

2. Create an enum for traffic signals (Red, Yellow, Green). Write a function that takes an enum value as input and prints its meaning (e.g., Red → Stop).

3. Write a C program to store and process student data using dynamic memory allocation with calloc(). The program should:

- Allow the user to enter the number of students (n).
- Dynamically allocate memory for n students using calloc().
- For each student, input details such as **roll number, name, and marks**.
- Display the list of students who have **failed** (consider marks less than 40 as failed).

4. Write a C program to print the corresponding weekday name for a given integer value (1 to 7) using enumeration constants. The program should define an enum for the weekdays, take an integer input from the user, and display the corresponding weekday name. If the input is out of range, display an appropriate error message.

5. Write a C program to write and read text into a file.The program should prompt the user to enter a string and write it to a text file. Then, it should open the same file, read the content, and display it on the screen.

6. Write a C program to copy the contents of one file to another using command-line arguments. The program should accept the source filename and the destination filename as command-line arguments. It should read the contents from the source file and write them to the destination file. If the source file does not exist or an error occurs, display an appropriate error message.

Web Resources:

1. LinuxCommand.org, "Basic Linux Environment, Editors, Turbo C, GCC," [Online]. Available: https://linuxcommand.org/. [Accessed: Feb. 25, 2025].
2. Programiz, "Pattern Printing & Block Letters in C," [Online]. Available: https://www.programiz.com/c-programming/examples/pattern-printing. [Accessed: Feb. 25, 2025].
3. GeeksforGeeks, "Arithmetic Operations & Swapping Numbers," [Online]. Available: https://www.geeksforgeeks.org/swap-two-numbers-without-using-temporary-variable/. [Accessed: Feb. 25, 2025].
4. CodingCompiler, "Sum and Average of Numbers," [Online]. Available: https://codingcompiler.com/c-program-find-sum-average-three-numbers/. [Accessed: Feb. 25, 2025].
5. TutorialsPoint, "Reading and Printing Multiple Integers," [Online]. Available: https://www.tutorialspoint.com/cprogramming/c_arrays.htm. [Accessed: Feb. 25, 2025].
6. Programiz, "Mixed Data Types Input & Format Specifiers," [Online]. Available: https://www.programiz.com/c-programming/c-input-output. [Accessed: Feb. 25, 2025].
7. GeeksforGeeks, "Files in C," [Online]. Available: https://www.geeksforgeeks.org/basics-file-handling-c/. [Accessed: Feb. 25, 2025].

24UC181 Design Thinking

Category: Engineering Sciences (ES) **0L 0T 2P 1C**

***Pre-requisite*: NIL**

Course Description:

This course introduces first-year undergraduate engineering students to the principles and practices of design thinking, emphasizing user-centred problem-solving and innovation. Students will learn to empathize with users, define problems, ideate solutions, develop prototypes, and test these solutions. The course will also explore the application of design thinking across various engineering disciplines, including Civil Engineering (CE), Electronics and Communication Engineering (EC), Electrical Engineering (EE), Mechanical Engineering (ME), and Information Technology (IT).

Teaching Objectives:

1. To introduce the fundamental concepts and process of design thinking.
2. To develop students' ability to empathize with users and accurately define engineering problems.
3. To encourage creativity and the generation of a wide range of ideas.
4. To enable students to prototype and test their design solutions.
5. To illustrate the application of design thinking to solve core engineering design problems.

Course Outcomes: At the end of the course, the student will be able to

1. explain the principles of design thinking. **[K2]**
2. apply empathy and problem-defining techniques in engineering contexts. **[K3]**
3. employ innovative ideas using various ideation techniques. **[K3]**
4. model and test prototypes to validate engineering solutions. **[K4]**
5. practice design thinking strategies in civil, electronic, electrical, mechanical, and IT engineering projects. **[K3]**

Course Contents

Unit 1: Introduction to Design Thinking **[2 hrs]**

Description: This unit introduces the foundational concepts of design thinking and its importance in solving complex engineering problems.

Lecture Topics:

1. Introduction to Design Thinking
2. History and Evolution of Design Thinking
3. Key Principles and Mindsets
4. The Design Thinking Process: An Overview
5. The Role of Design Thinking in Engineering

Examples/Applications/Case Studies:

- Case Study: How IDEO uses design thinking to innovate.
- Example: Apple's design thinking approach in product development.

Lab/Practice Exercises:

1. Identifying and analyzing engineering problems.
2. Group discussions on real-world design thinking examples.
3. Role-playing to understand user perspectives.

Learning Outcomes:

1. Understand the basic principles of design thinking.
2. Articulate the importance of design thinking in engineering.

Digital Resources:

- **Title:** Design Thinking: A Beginner's Guide
  Author(s): Interaction Design Foundation
  Year of Publication: 2020
  Publisher: Interaction Design Foundation
  URL: Design Thinking

Unit 2: Empathize and Define **[3 hrs]**

Description: This unit focuses on understanding users through empathy and accurately defining engineering problems to guide the design process.

Lecture Topics:

1. The Importance of Empathy in Design Thinking
2. Techniques for Empathy: Interviews, Observations, and Surveys
3. Defining the Problem: Point of View Statements
4. Tools for Problem Definition: Journey Maps, Personas
5. Translating Empathy into Insights

Examples/Applications/Case Studies:

- Example: Empathy maps in understanding customer needs.
- Case Study: How Airbnb uses empathy in design thinking.

Lab/Practice Exercises:

1. Conducting user interviews.
2. Creating personas and journey maps.
3. Developing problem statements.

Learning Outcomes:

1. Develop empathy to understand user needs.
2. Define clear and actionable engineering problem statements.

Digital Resources:

- **Title:** The Field Guide to Human-Centered Design
  Author(s): IDEO.org
  Year of Publication: 2015
  Publisher: IDEO.org
  URL: IDEO Field Guide

Unit 3: Ideate [3 hrs]

Description: This unit explores creative techniques for generating a wide range of ideas to solve defined engineering problems.

Lecture Topics:

1. The Ideation Phase in Design Thinking
2. Brainstorming Techniques
3. Creative Thinking and Innovation
4. Selecting and Refining Ideas
5. Overcoming Creative Blocks

Examples/Applications/Case Studies:

- Example: Brainstorming session in a tech startup.
- Case Study: Google's "20% time" policy for innovation.

Lab/Practice Exercises:

1. Group brainstorming sessions.
2. Idea selection and refinement exercises.
3. Sketching and visualizing ideas.

Learning Outcomes:

1. Generate innovative ideas through various brainstorming techniques.
2. Refine and select feasible engineering ideas.

Digital Resources:

- **Title:** Creative Confidence: Unleashing the Creative Potential Within Us All
  Author(s): Tom Kelley, David Kelley
  Year of Publication: 2013
  Publisher: Crown Business
  URL: Creative Confidence

Unit 4: Prototype and Test [3 hrs]

Description: This unit covers the creation of prototypes and the methods for testing and iterating on these prototypes to develop viable engineering solutions.

Lecture Topics:

1. Prototyping Basics and Importance
2. Types of Prototypes: Low-fidelity vs. High-fidelity
3. Tools and Materials for Prototyping
4. Testing Prototypes: Methods and Techniques
5. Iterating Based on Feedback

Examples/Applications/Case Studies:

- Example: Prototyping in the automotive industry.
- Case Study: Iterative prototyping at Dyson.

Lab/Practice Exercises:

1. Building low-fidelity prototypes.
2. Conducting user testing sessions.
3. Analyzing feedback and iterating designs.

Learning Outcomes:

1. Develop and test prototypes effectively.
2. Iterate on designs based on user feedback.

Digital Resources:

- **Title:** Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days
  Author(s): Jake Knapp, John Zeratsky, Braden Kowitz
  Year of Publication: 2016
  Publisher: Simon & Schuster
  URL: Sprint Book

Unit 5: Application of Design Thinking in Engineering [3 hrs]

Description: This unit examines the application of design thinking across various engineering disciplines, including CE, EC, EE, ME, and IT, to solve engineering design problems.

Lecture Topics:

1. Design Thinking in Civil Engineering (CE)
2. Applying Design Thinking in Electronics and Communication Engineering (EC)
3. Design Thinking for Electrical Engineering (EE)
4. Innovation in Mechanical Engineering (ME)
5. Design Thinking in Information Technology (IT)

Examples/Applications/Case Studies:

- Case Study: Design thinking in sustainable building design (CE).
- Example: Innovation in wearable technology (EC).
- Case Study: Smart grid solutions (EE).
- Example: Prototyping in automotive design (ME).
- Case Study: User-centric software development (IT).

Lab/Practice Exercises:

1. Developing design thinking strategies for a civil engineering project.
2. Prototyping an electronic device solution.
3. Creating user-centric IT solutions.

Learning Outcomes:

1. Apply design thinking principles to various engineering disciplines.
2. Develop innovative solutions in CE, EC, EE, ME, and IT.

Digital Resources:

- **Title:** Design a Better Business: New Tools, Skills, and Mindset for Strategy and Innovation
  Author(s): Patrick Van Der Pijl, Justin Lokitz, Lisa Kay Solomon
  Year of Publication: 2016
  Publisher: Wiley
  URL: Design a Better Business

Textbooks

1. Brown, T. (2009). *Change by Design: How Design Thinking Creates New Alternatives for Business and Society*. Harper Business.
2. Cross, N. (2011). *Design Thinking: Understanding How Designers Think and Work*. Berg.

Reference Books

1. Liedtka, J., & Ogilvie, T. (2011). *Designing for Growth: A Design Thinking Tool Kit for Managers*. Columbia University Press.
2. Kelley, T., & Kelley, D. (2013). *Creative Confidence: Unleashing the Creative Potential Within Us All*. Crown Business.

3. Martin, R. L. (2009). *The Design of Business: Why Design Thinking is the Next Competitive Advantage*. Harvard Business Press.

Web Resources

1. **IDEO Design Thinking** - IDEO's official website provides numerous resources and case studies on design thinking practices and applications.
    - https://www.ideo.com
2. **d.school: Institute of Design at Stanford** - Stanford's d.school offers a comprehensive range of materials, including articles, guides, and toolkits on design thinking.
    - https://dschool.stanford.edu/resources
3. **Interaction Design Foundation** - A website offering various articles and educational materials on design thinking and related concepts.
    - https://www.interaction-design.org/literature/topics/design-thinking

24PH181 PHYSICS LAB

***Category*: Basic Sciences (BS)** **0L 0T 2P 1C**

***Pre – requisite*: 10 + 2 Physics**

Course Description:

The Engineering Physics laboratory for B. Tech students is designed to provide hands on experience in basic concepts of optics, lasers, optical fibers, waves and oscillators, electricity and magnetism, quantum mechanics, and semiconductors. This course aims to foster the practical skills essential through experimentation, measurement, and analysis for the students of computer science allied branches like CSE, AI-ML, AI-DS, and Information Technology, ECE, EEE, EIE, CE and ME branches. Students will work with advanced equipment under the guidance of experienced faculty to explore various physical phenomena and their engineering applications.

Course Objectives:

1. Demonstrate the basic concepts of wave optics and the experimental evidence of wave nature of light by interference and diffraction phenomena.
2. Provide the experimental knowledge of understanding the properties of semiconductors and their applications.
3. Elucidate the basic concepts of Lasers and Optical fibers and their diverse applications in Science and Technology.

Course Outcomes:

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

1. identify the wave nature of light by the concepts like interference and diffraction of light (K4)
2. distinguish the semiconductors based on carrier concentration and Hall coefficient (K4)
3. calculate the specific charge of an electron and work function of a photocell (K4)
4. demonstrate a comprehensive understanding of wave phenomena, resonance, and their application across various physical systems (K3)
5. apply the laser principles, optical fiber transmission characteristics, and their practical applications in telecommunications and photonics (K4)

Course Content:

Experiments common to all branches:

1. Solar cell – Determination of fill factor (Common to all branches).
2. Hall effect – Hall coefficient measurement (Common to all branches).

Experiments for CSE, IT, AIDS, and AIML branches:

3. Newton's Rings – Radius of Curvature of a plano convex lens
4. Diffraction grating – Wavelength of laser light
5. Photocell – Study of V – I characteristics and determination of work function
6. Compound pendulum – Measurement of 'g'
7. Specific charge (e/m) of an electron – J. J. Thomson method
8. AC Sonometer – Verification of vibrating laws
9. Optical Fiber – Determination of Numerical Aperture
10. Figure of Merit of a Galvanometer
11. Variation of Magnetic field along the axis of current – carrying circular coil
12. Diffraction grating – Measurement of wavelength of mercury source

Experiments for ECE, EEE, and EIE branches:

3. Figure of Merit of a Galvanometer.
4. LCR circuit – Study of Resonance.
5. Variation of Magnetic field along the axis of current – carrying circular coil
6. Wedge method – Measurement of thickness of a foil.
7. Specific charge (e/m) of an electron – J. J. Thomson method
8. B-H Curve Unit- Determination of hysteresis loss
9. Diffraction grating – Wavelength of laser light
10. Photocell – Study of V – I characteristics, determination of work function.
11. Optical Fiber – Determination of Numerical Aperture
12. Torsional pendulum-Measurement of Rigidity Modulus

Experiments for CE and ME branches:

3. AC Sonometer – Verification of vibrating laws
4. Wedge method – Measurement of thickness of a foil
5. Diffraction grating – Wavelength of laser light Optical Fiber
6. Photocell – Study of V – I characteristics and determination of work function
7. Torsional pendulum-Measurement of Rigidity Modulus
8. Determination of Dielectric constant of a sample
9. Optical Fiber – Determination of Numerical Aperture
10. Compound pendulum – Measurement of 'g'
11. Variation of Magnetic field along the axis of current – carrying circular coil
12. Figure of Merit of a Galvanometer.

Text Books:

1. Panigrahi, S., & Mallick, B. (2015), *Engineering Practical Physics* (1st ed.). Cengage Learning.
2. Madhusudhana Rao, C. V., & Vasanth Kumar, V. (2015), *Engineering Physics Lab Manual* (4th ed.). Scitech publications.

Web Resources:

1. https://phet.colorado.edu/en/simulations/wave-interference.
2. http://nationalmaglab.org/magnet-academy/
3. https://wanda.fiu.edu/boenlinw/courses/modern_lab_manual3/em_ratio.html
4. https://www.learnabout-electronics.org/ac_theory/lcr_series.php
5. https://www.sciencefacts.net/hall-effect.html
6. https://lipa.physics.oregonstate.edu/sec_diffraction-gratings.html
7. https://academy.cba.mit.edu/classes/input_devices/meas.pdf
8. https://cdac.olabs.edu.in/?sub=1&brch=6&sim=151&cnt=2

Virtual Lab References:

1. http://vlab.amrita.edu/index.php?sub=1&brch=192&sim=972&cnt=4
2. http://vlab.amrita.edu/index.php?sub=1&brch=189&sim=335&cnt=4
3. http://vlab.amrita.edu/index.php?sub=1&brch=280&sim=210&cnt=4
4. http://vlab.amrita.edu/index.php?sub=1&brch=281&sim=334&cnt=1
5. http://vlab.amrita.edu/index.php?sub=1&brch=74&sim=1523&cnt=4
6. http://vlab.amrita.edu/index.php?sub=1&brch=75&sim=330&cnt=4
7. http://vlab.amrita.edu/index.php?sub=1&brch=282&sim=1507&cnt=4
8. http://vlab.amrita.edu/index.php?sub=1&brch=280&sim=1518&cnt=4
9. http://vlab.amrita.edu/index.php?sub=1&brch=201&sim=366&cnt=4
10. http://vlab.amrita.edu/index.php?sub=1&brch=195&sim=840&cnt=4

24UC182 AI Tools and Applications

Category: Engineering Sciences (ES) **0L 0T 2P 1C**

***Pre-requisite*: NIL**

Course Description:

This course introduces undergraduate engineering students to the fundamental concepts and applications of Artificial Intelligence (AI). The course covers a range of topics from the basics of AI to its use as a personal assistant, learning tool, and generative AI for literature review and writing. Additionally, the course delves into specific AI applications in Civil Engineering (CE), Electronics and Communication Engineering (EC), Information Technology (IT), and Mechanical Engineering (ME).

Course Objectives:

1. To provide an overview of AI and its applications in various engineering fields.
2. To demonstrate the use of AI as a personal assistant for productivity enhancement.
3. To explore AI as a tool for logical reasoning and mathematical analytics.
4. To utilize generative AI for literature review, gap analysis, ideation, and article writing.
5. To discuss detailed applications of AI in CE, EC, IT, and ME.

Course Outcomes: By the end of the course, students will be able to:

1. explain the basic concepts of AI and its historical evolution [K2].
2. apply AI tools for productivity enhancement [K3].
3. develop logical reasoning and analytical skills using AI tools [K3].
4. conduct literature reviews and write articles using generative AI [K3].
5. implement AI solutions in their respective engineering branches [K3].

Course Structure

Unit 1: Introduction to AI & Applications

Description: This unit introduces the basic concepts, history, and various applications of AI. It lays the foundation for understanding how AI has evolved and where it is applied in today's world.

Contents:

- Definition and History of AI
- Evolution of AI: From Early Concepts to Modern Developments
- Types of AI: Narrow AI, General AI, and Super AI
- AI Applications in Industry and Daily Life
- Ethical and Societal Implications of AI

Examples/Applications/Case-studies:

- AI in Search Engines: Google's AI Algorithms
- AI in Healthcare: Disease Diagnosis and Treatment Plans
- AI in Finance: Automated Trading Systems
- AI in Customer Service: Chatbots

Exercise/Project Problems:

1. **Research and Present the Evolution of AI:** Create a timeline detailing significant milestones in the development of AI from its inception to the present.
2. **Ethical Implications of AI:** Write a report analyzing the ethical considerations and potential societal impacts of AI in autonomous vehicles.
3. **Develop a Simple AI Chatbot:** Using available online tools, create a chatbot that can answer basic questions and simulate a conversation.

Tool Studies:

- Introduction to Python Programming
- Overview of AI Libraries: TensorFlow, Keras

Practical/Lab Exercises:

1. **Set Up Python Environment:** Install Python and set up the necessary libraries to run basic AI programs.
2. **Image Recognition with Pre-trained Models:** Use a pre-trained model to perform image recognition tasks.

Learning Outcomes:

- Understand the historical context and evolution of AI
- Identify various AI applications in different sectors

References/E-Resources:

- "Artificial Intelligence: A Modern Approach," Stuart Russell and Peter Norvig, 2020, Pearson. URL

Unit 2: AI as a Personal Assistant for Productivity Enhancement

Description:
This unit explores the use of AI as a personal assistant to improve productivity in various aspects of daily life and work.

Contents

- AI-driven Personal Assistants (e.g., Siri, Google Assistant)
- AI in Task Management and Scheduling
- AI in Email Management and Communication
- AI for Personal Productivity: Time Management, Reminder Systems

Examples/Applications/Case-studies:

- Automating Daily Schedules with Google Assistant
- AI for Sorting and Prioritizing Emails
- AI Tools for Managing Personal Tasks and Deadlines

Exercise/Project Problems:

1. **Task Management App:** Create a task management application using AI to prioritize and schedule tasks based on user input and deadlines.
2. **Email Assistant:** Develop an AI system to filter and prioritize emails based on content and importance.
3. **Voice-Activated Assistant:** Build a voice-activated assistant capable of performing basic commands such as setting reminders and providing weather updates.

Tool Studies:

- Dialogflow for Creating Voice Assistants
- Alexa Skills Kit for Developing Voice Apps

Practical/Lab Exercises:

1. **Create a Voice Assistant Using Dialogflow:** Develop a simple voice assistant that can perform basic tasks.
2. **Email Sorting Algorithm:** Implement an AI model to automatically sort and prioritize emails based on user-defined criteria.

Learning Outcomes:

- Apply AI tools to enhance personal productivity
- Develop AI-based applications for personal assistant functions.

References/E-Resources:

- "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow," Aurélien Géron, 2019, O'ReillyMedia. URL

Unit 3: AI as a Learning Tool for Logical Reasoning & Mathematical Analytics

Description:
This unit focuses on using AI tools to improve logical reasoning and perform complex mathematicalanalytics, enhancing problem-solving skills.

Contents

- AI in Problem-Solving and Decision-Making
- AI in Mathematical Modeling and Simulation
- AI-Driven Analytical Tools for Data Analysis
- Enhancing Logical Reasoning with AI

Examples/Applications/Case-studies:

- AI in Financial Analytics for Predicting Market Trends
- AI in Scientific Research for Analyzing Large Data Sets
- AI Models for Complex Problem-Solving in Engineering

Exercise/Project Problems:

1. **Solve Logical Puzzles with AI:** Use AI to develop solutions for complex logical puzzles and games.
2. **Financial Data Analysis:** Implement AI models to analyze and predict financial market trends.
3. **Simulation Model:** Create a simulation model using AI to address a specific problem scenario.

Tool Studies:

- MATLAB for Mathematical Modeling
- R and Python Libraries for Data Analytics

Practical/Lab Exercises:

1. **Mathematical Modeling in MATLAB:** Use MATLAB to create and analyze a mathematical model based on given data.
2. **Decision-Making Algorithm:** Implement a basic AI algorithm for decision-making using Python.

Learning Outcomes:

- Analyze and solve problems using AI tools
- Develop AI models for mathematical and logical applications.

References/E-Resources:

- "Mathematics for Machine Learning," Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong, 2020,Springer. URL

Unit 4: Generative-AI for Literature-Review, Gap-Analysis, Ideation, and Article-Writing

Description:
This unit explores the application of generative AI tools for academic and research purposes, focusing on literature review, gap analysis, ideation, and writing.

Contents

- Introduction to Generative AI
- Using AI for Literature Review and Research
- AI for Identifying Research Gaps and Ideation
- AI in Academic and Article Writing

Examples/Applications/Case-studies:

- Utilizing GPT-3 for Generating Research Ideas
- AI Tools for Automated Literature Reviews
- Writing Articles and Reports Using Generative AI

Exercise/Project Problems:

1. **Conduct a Literature Review:** Use AI tools to conduct a comprehensive literature review on a chosen topic.
2. **Identify Research Gaps:** Employ AI to identify gaps in existing research within a specific field.
3. **Write a Research Article:** Use generative AI tools to draft a research article on a given topic.

Tool Studies:

- OpenAI GPT-3 for Content Generation
- Scite.ai for Research and Literature Review

Practical/Lab Exercises:

1. **Generate Research Ideas with GPT-3:** Use GPT-3 to generate and refine research ideas for a project.
2. **Automated Literature Review:** Conduct a literature review using AI tools and summarize key findings.

Learning Outcomes:

- Utilize generative AI for academic research and writing.
- Conduct gap analysis and ideation using AI tools.

References/E-Resources:

- "Deep Learning," Ian Goodfellow, Yoshua Bengio, and Aaron Courville, 2016, MIT Press. URL

Unit 5: AI Applications in Engineering

Description:
This unit provides an in-depth exploration of AI applications in Civil Engineering (CE), Electronics and Communication (EC), Information Technology (IT), and Mechanical Engineering (ME).

Contents

- AI in Civil Engineering: Smart Cities, Structural Health Monitoring
- AI in Electronics and Communication: Signal Processing, Communication Systems
- AI in Information Technology: Cybersecurity, Software Development
- AI in Mechanical Engineering: Robotics, Predictive Maintenance

Examples/Applications/Case-studies:

- Predictive Maintenance in Manufacturing using AI
- Smart Grid Management with AI
- AI in Traffic Management for Smart Cities
- AI in Communication Systems for Signal Processing

Exercise/Project Problems:

1. **Predictive Maintenance Model:** Develop an AI model for predictive maintenance in a mechanical system.
2. **AI in Signal Processing:** Implement AI-based signal processing techniques in communication systems.
3. **Smart City Simulation:** Create a simulation of a smart city using AI for traffic management and infrastructure optimization.

Tool Studies:

- ANSYS for Civil Engineering Simulations
- MATLAB for Signal Processing in Electronics
- Various AI tools for IT and Mechanical Engineering applications

Practical/Lab Exercises:

1. **Predictive Maintenance System:** Develop a simple AI-based predictive maintenance system using Python and relevant libraries.
2. **Signal Processing Algorithm:** Implement an AI algorithm for signal processing tasks using MATLAB.

Learning Outcomes:

- Apply AI tools and techniques in specific engineering fields.
- Develop AI-based solutions tailored to engineering problems.

References/E-Resources:

- "Artificial Intelligence and Machine Learning for Engineers," Tarun Kumar Sharma, 2020, Springer. URL

Detailed Projects for Students:

1. **Smart Traffic Management System:**
   - **Objective:** Develop an AI-based system to manage traffic flow in a smart city.
   - **Description:** The project involves collecting data from traffic sensors, performing real-time traffic analysis, and implementing adaptive traffic signal control to optimize traffic flow and reduce congestion.
   - **Tools:** Python, TensorFlow, Keras, traffic simulation tools.
2. **AI-Based Predictive Maintenance for Manufacturing:**
   - **Objective:** Create an AI system to predict equipment failures in a manufacturing plant.
   - **Description:** The project includes data collection from sensors, data analysis to predict potential equipment failures, and a maintenance scheduling system to prevent downtime.
   - **Tools:** Python, TensorFlow, Keras, IoT sensors.
3. **AI-Driven Cybersecurity System:**
   - **Objective:** Implement an AI system for detecting and mitigating cybersecurity threats.
   - **Description:** The project involves analyzing network traffic data to detect anomalies, identifying potential threats using machine learning algorithms, and developing automated responses to mitigate these threats.
   - **Tools:** Python, Scikit-learn, TensorFlow, network monitoring tools.

Textbooks/References:

1. Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.
2. Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media.
3. Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for Machine Learning*. Springer.
4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
5. Sharma, T. K. (2020). *Artificial Intelligence and Machine Learning for Engineers*. Springer.

24UC101 Essence of Indian Knowledge Tradition

Course Category: Mandatory (MC) 2L 0T 0P 0C

Pre-requisites: Nil

Course Description:

This course provides an in-depth exploration of the rich Indian knowledge tradition spanning philosophy, scientific advancements, cultural perspectives, educational heritage, and contemporary relevance. Students will delve into foundational texts, philosophical schools, scientific contributions, cultural artefacts, and the impact of Indian knowledge systems on global thought.

Teaching Objectives:

1. To introduce students to the major philosophical schools (Darshanas) of India and their foundational principles.
2. To familiarize students with significant scientific contributions in mathematics, astronomy, medicine, and technology from ancient India.
3. To explore the cultural perspectives of India through its literature, arts, architecture, and political systems.
4. To understand the educational heritage of ancient India and its impact on holistic learning and societal evolution.
5. To analyse the influence and contemporary relevance of Indian knowledge systems globally.

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

CO1: analyse and differentiate between the various philosophical schools of India.

CO2: evaluate the contributions of ancient Indian scientists and their impact on global knowledge systems.

CO3: interpret and discuss the cultural and societal norms depicted in Indian literature, arts, and architecture.

CO4: critically assess the educational methods and institutions of ancient India and their relevance today.

CO5: discuss the challenges and opportunities in reviving and adapting traditional Indian knowledge systems in contemporary contexts.

Course Structure:

Unit 1: Foundations of Indian Philosophy

- **Unit Description:** Introduction to Darshanas: Nyaya, Vaisheshika, Samkhya, Yoga, Mimamsa, Vedanta.
- **Topics:** Introduction to Darshanas; Detailed study of Nyaya, Vaisheshika, Samkhya, Yoga, Mimamsa, and Vedanta.
- **Examples/Applications/Case-studies:** Comparative analysis of philosophical concepts; Case studies of ethical dilemmas from Nyaya and Mimamsa perspectives.
- **Learning Outcomes:** Understand and articulate the fundamental concepts of each Darshana; Apply philosophical principles to modern ethical issues.

Unit 2: Scientific Advancements

- **Unit Description:** Contributions of Aryabhata, Brahmagupta, and others in Mathematics, Astronomy, Medicine, Ayurveda, Metallurgy, Chemistry, and Technology.
- **Topics:** Mathematical theorems of Aryabhata; Astronomical discoveries of Brahmagupta; Principles of Ayurveda; Techniques in Metallurgy and Chemistry.
- **Examples/Applications/Case-studies:** Application of Aryabhata's astronomy in navigation; Ayurvedic treatments for common ailments.
- **Learning Outcomes:** Analyze the scientific contributions of ancient Indian scholars; Apply historical techniques to practical scenarios.

Unit 3: Cultural Perspectives

- **Unit Description:** Epics (Ramayana, Mahabharata), classical texts (Puranas, Shastras); Arts and architecture; Political systems, governance, ethics, and societal norms.
- **Topics:** Analysis of Ramayana and Mahabharata; Study of classical texts; Evolution of temple architecture; Ethical governance principles.
- **Examples/Applications/Case-studies:** Comparative study of epic narratives across cultures; Architectural case studies.
- **Learning Outcomes:** Interpret cultural artefacts; Critique societal norms depicted in literature and architecture.

Unit 4: Educational Heritage of Ancient India

- **Unit Description:** Holistic learning in nature; Universities and specialization; Education for women; Knowledge exchange beyond India.
- **Topics:** Gurukul system; Nalanda and Takshashila universities; Role of women scholars; Influence on Southeast Asian education.
- **Examples/Applications/Case-studies:** Impact of Nalanda teachings on Asian philosophies; Comparative analysis of ancient and modern educational methods.
- **Learning Outcomes:** Assess the educational legacy of ancient India; Propose reforms based on historical insights.

Unit 5: Influence and Contemporary Relevance

- **Unit Description:** Impact on global thought; Contemporary interpretations and adaptations; Challenges in reviving traditional knowledge systems.
- **Topics:** Spread of Indian philosophies to the West; Modern adaptations of Ayurveda; Challenges in preserving oral traditions.
- **Examples/Applications/Case-studies:** Case studies of global adoption of yoga; Challenges in integrating traditional and modern medical practices.
- **Learning Outcomes:** Evaluate the global impact of Indian knowledge; Propose strategies for preserving and adapting traditional knowledge.

Textbooks/References:

- Radhakrishnan, S., & Moore, C. A. (Eds.). (1957). *A Source Book in Indian Philosophy.* Princeton University Press.
- Basham, A. L. (1954). *The Wonder That Was India: A Survey of the Culture of the Indian Sub-Continent Before the Coming of the Muslims.* Sidgwick & Jackson.

e- Resources:

Indian Philosophy: Indian Philosophy | Internet Encyclopedia of Philosophy (utm.edu)

Aryabhata: Aryabhata | Achievements, Biography, & Facts | Britannica

Ramayana and Mahabharata: The Ramayana and Mahabharata Index (sacred-texts.com)

Nalanda Univ: Nalanda University | Home - Nalanda University

Gurukul System: Gurukul education system in ancient india - Vidhyanjali Academy School

How Indian Philosophies Influenced Western Academia: The Dharma of Intellectualism (substack.com)