RAMDEOBABA UNIVERSITY, NAGPUR–440013

Established by the Maharashtra Private Universities (Establishment and Regulation) Act 2023 (Mah. Act No VIII of 2024)
Formerly, Shri Ramdeobaba College of Engineering and Management, Nagpur 440013

School of Electrical and Electronics Engineering

Department of Electronics Engineering

PROGRAMME SCHEME & SYLLABI

of First Year as per National Education Policy (NEP)

(With effect from Academic Year 2025-26)

B.Tech. ELECTRONICS AND COMPUTER SCIENCE powered by IBM

Semester I

SN	Course Type	Code	Course	Hours/ week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	P	C						
1	ESC	25EE01TP0101	Basic Electronics and Computer Fundamentals	3	2	4	20/25	30	50	25	150	3
2	BSC	25HS03TH0103	Probability and statistics	3	0	3	20	30	50	-	100	3
3	ESC	25EE01TP0102	Digital Logic Design	3	2	4	20/25	30	50	25	150	3
4	PCC	25EE01TP0103	Python Programming	3	2	4	20/25	30	50	25	150	3
5	CCA	25EE01TP0104	Design Thinking and Innovation	1	2	2	20/25	30	-	25	100	
6	VSEC	25EE01PR0105	Computer Workshop	0	2	1	25	-	-	25	50	
7	CCA	25HS02PR0102-18	Liberal / Performing Art Lab	0	2	1	25	-	-	25	50	
TOTAL				**13**	**12**	**19**						

Semester II

SN	Course Type	Code	Course	Hours / week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	P	C						
1	ESC	25EE01TP0201	Microcontroller and Interfacing	3	2	4	20/25	30	50	25	150	3
2	BSC	25HS03TH0213	Calculus & Linear Algebra	3	0	3	20	30	50	-	100	3
3	PCC	25EE01TP0202	Programming for Problem Solving	3	2	4	20/25	30	50	25	150	3
4	ESC	25EE01TH0203	AI and ML Essentials	3	0	3	20	30	50	-	100	3
5	ESC	25EE01TP0204	Computer Architecture and Organization	3	2	4	20/25	30	50	25	150	3
6	AEC	25HS02TP0201	English for Professional Communication	2	2	3	20/25	30	50	25	150	2
7	IKS	25HS02TH0203-1	Foundational Literature of Indian Civilization	1	0	1	20	30	--	--	50	
8	VSEC	25EE01PR0205	Mini Project-I	0	2	1	25	-	-	25	50	
9	CCA	25HS04PR0201	Health-Fitness-Wellbeing (HFW)	0	2	1	25	-	-	25	50	
TOTAL				**18**	**12**	**24**						

`

Semester III

SN	Course Type	Code	Course	Hours/ week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	P	C						
1	PCC	25EE01TP0301-1 OR 25EE01TP0301-2	Object oriented programming using Python OR Object oriented programming using JAVA	3	2	4	20/25	30	50	25	150	3
2	PCC	25EE01TP0302	IoT Fundamentals and Architecture	3	2	4	20/25	30	50	25	150	3
3	PCC	25EE01TP0303	Machine Learning Engineering	3	2	4	20/25	30	50	25	150	3
4	PCC	25EE01TP0304	Digital Signal Processing	3	2	4	20/25	30	50	25	150	3
5	MDM	25EE01TH0305	MDM-1	3	0	3	20	30	50	-	100	3
	OE	25EEOEC01TH0306	Open Elective- I / MOOCs / Industry Offered	2	0	2	20	30	50	-	100	2
6	VSEC	25EE01PR0307	Mini Project II	0	4	2	25	-	-	25	50	
7	VEC	25HS02TH0301	Foundational course in Universal Human	1	0	1	20	30	-	-	50	

			Value									
TOTAL				**18**	**12**	**24**						

Semester IV

	Course Type	Code	Course	Hours / week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	P	C						
1	ESC	25EE01TP0401	Deep Learning	3	2	4	20/25	30	50	25	150	3
2	PEC	25EE01TP0402	Program Elective I	3	2	4	20/25	30	50	25	150	3
3	PCC	25EE01TP0403-1 OR 25EE01TP0403-2	Data Structures and Algorithms using Python OR Data Structures and Algorithms using JAVA	3	2	4	20/25	30	50	25	150	3
4	AEC	25EE01PR0404	Competitive coding-I	0	2	1	25	-	-	25	50	
5	MDM	25EE01TH0405	MDM-II	3	0	3	20	30	50	-	100	3
6	OE	25EEOEC01TH 0406	Open Elective- II / MOOCs / Industry Offered	2	0	2	20	30	50	-	100	2
7	VSEC	25EE01PR0407	Mini Project III	0	4	2	25	-	-	25	50	
8	CCA	25HS04PR0401	Self Defence and Indian Martial Art	0	2	0	25	-	-	25	50	
9	VSEC	25EE01PR0408	Aptitude Development	0	2	1	25	-	-	25	50	
TOTAL				**14**	**16**	**21**						

Semester V

SN	Course Type	Code	Course	Hours /week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	P	C						
1	PCC	25EE01TP0501	Database Management System	3	2	4	20/25	30	50	25	150	3
2	PEC	25EE01TP0502	Program Elective-II	3	2	4	20/25	30	50	25	150	3
3	PEC	25EE01TP0503	Program Elective-III	3	2	4	20/25	30	50	25	150	3
4	PCC	25EE01TP0504	Operating System	3	2	4	20/25	30	50	25	150	3
5	VSEC	25EE01PR0505	Mini Project IV	0	4	2	25	-	-	25	50	
6	OE	25EEOEC01TH 0506	Open Elective-III / MOOCs / Industry Offered	2	0	2	20	30	50	-	100	2
7	MDM	25EE01TH0507	MDM-III	3	0	3	20	30	50	-	100	3
TOTAL				**17**	**12**	**23**						

Semester VI

SN	Course Type	Code	Course	Hours/ week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	**P**	**C**						
1	PEC	25EE01TP0601	Program Elective-IV	3	2	4	20/25	30	50	25	150	3
2	PEC	25EE01TP0602	Program Elective-V	3	2	4	20/25	30	50	25	150	3
3	PCC	25EE01TP0603	Computer Networks	3	2	4	20/25	30	50	25	150	3
4	PCC	25EE01PR0604	Digital VLSI Design	3	2	4	20/25	30	50	25	150	3
5	MDM	25EE01TH0605	MDM-IV	3	0	3	20	30	50	-	100	3
6	AEC	25EE01PR0607	Competitive coding-II	0	2	1	25	-	-	25	50	
7	HSSM	25HS02TP0501	Business Communication	1	2	2	20/25	30	-	25	100	--
TOTAL				**16**	**12**	**22**						

Semester VII

SN	Course Type	Code	Course	Hours/ week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	P	C						
1	PCC	25EE10TP0701	Cloud Computing	3	2	4	20/25	30	50	25	150	3
2	PCC	25EE10TP0702	Information Security and Cryptography	3	2	4	20/25	30	50	25	150	3
3	PCC	25EE10TH0703	Blockchain	2	0	2	20	30	50	-	100	2
4	PRJ	25EE10PR0704	Capstone Project-I	0	8	4	25	-	-	25	50	
5	FP	25EE10PR0705	Internship Evaluation	0	2	0	25	-	-	25	50	
6	AEC	25EE10PR0706	Participative Learning	0	2	1	25	-	-	25	50	
TOTAL				**8**	**16**	**15**						

Semester VIII

SN	Course Type	Code	Course	Hours / week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	**P**	**C**						
1	PCC	25EE01TH0801	Cognitive & Behavioral Skills for Engineers / Parallel and Distributed computing / MOOCs	3	0	3	20	30	50	-	100	3
2	VEC	25EE01TH0802	Cyber Laws and Ethics in IT / MOOCs	3	0	3	20	30	50	-	100	2
3	PRJ	25EE01PR0803	Capstone Project-II	0	12	6	50	-	-	50	100	
TOTAL				**6**	**12**	**12**						
OR												
1	Internship	24EE01PR0804	Industry Internship / TBI Internship / Research Internship	0	24	**12**	50	-	-	50	100	
				0	**24**	**12**						

For one-year full time internship

SN	Course Type	Code	Course	Hours / week			Internal Evaluation (Th/Lab)	Mid Sem Exam	End Sem Exam	Continuous Evaluation	Total Marks	ESE Duration (Hrs)
				L	P	C						
1	Internship	24EE01PR0805	Industry Internship / TBI Internship	0	48	**24**	**50**	-	-	50	100	
2	PCC	24EE01TH0806	MOOCs	3	0	**3**						
Total				3	48	**27**						

HONORS Specialization in Research

SN	Sem	Code	Course	Hours/ week		C	Internal Evaluation (Th/Lab)	Mid Sem Exam (30)	End Sem Exam (50)	Continuous Evaluation (25)	Total Marks
				L	P						
1	VII	25EE10HT0701	Research Methodology	3	0	3	20	30	50	-	100
2	VII	25EE10HP0702	Research Project Phase - I	0	6	3	25	-	-	25	50
3	VIII	25EE10HP0801	Research Project Phase-II	0	24	12	25	-	-	25	50
TOTAL				3	30	18					
				33 Hrs.							

Program Elective Basket

Specialization	Program Elective-I	Program Elective-II	Program Elective-III	Program Elective-IV	Program Elective-V
IBM Specialization	25EE01TP0402-1	25EE01TP0502-1	25EE01TP0503-1	25EE01TP0601-1	25EE01TP0602-1
	Natural Language Processing	Computer Vision	Applied Large Language Model Development and Deployment	Multimodal AI and Language-Visual Models	Explainable AI (XAI)
ECS1	25EE01TP0402-2	25EE01TP0502-2	25EE01TP0503-2	25EE01TP0601-2	25EE01TP0602-2
	Design and Development of Real-Time Embedded Systems	Embedded Linux and Driver Development	Distributed Edge Computing Systems and Applications	Security Architectures in Embedded and Cyber-Physical Systems	Designing Intelligent Applications at the Edge
ECS2	25EE01TP0402-3	25EE01TP0502-3	25EE01TP0503-3	25EE01TP0601-3	25EE01TP0602-3
	Digital System Design	SoC Design	Digital Design Verification	Physical Design Flow and Synthesis	High Level Synthesis Design
ECS3	25EE01TP0402-4	25EE01TP0502-4	25EE01TP0503-4	25EE01TP0601-4	25EE01TP0602-4
	Web Application Development	Backend Frameworks and API's	Database Integration with ORM	DevOps and cloud Deployment	Full stack system design

HONORS Specialization in Electronics and Computer Science (AI for edge computing)

Sr. No.	Semester	Course Code	Course Title	Hours/week			Credits	Maximum Marks			ESE Duration (Hrs.)
								Continuous Evaluation	End Sem Exam	Total	
				L	T	P					
1	III	25EE01HT0301	Fundamentals of Artificial intelligence and Edge Computing	3	0	0	3	50	50	100	3
2	IV	25EE01HT0401	Single dimension signal processing on Edge	3	0	0	3	50	50	100	3
3	V	25EE01HT0501	Image and Video Signal Processing on Edge	3	1	0	4	50	50	100	3
4	VI	25EE01HT0601	Computer vision with Embedded Machine Learning	3	1	0	4	50	50	100	3
5	VII	25EE01HP0701	Honors Project	0	0	8	4	50	50	100	-
				12	**2**	**8**	**18**				

MINOR Specialization in IoT

SN	**Sem**	**Code**	**Course**	**Hours/ week**			**Internal Evaluation (Th/Lab)**	**Mid Sem Exam**	**End Sem Exam**	**Continuous Evaluation**	**Total Marks**
				L	**P**	**C**					
1	III	25EE10MT0301	IoT fundamentals	3	0	3	20	30	50	-	100
2	IV	25EE10MT0401	Sensor Interfacing with Arduino and ESP8266	3	0	3	20	30	50	-	100
3	V	25EE10MT0501	Cloud Computing Using Raspberry Pi	4	0	4	20	30	50	-	100
4	VI	25EE10MT0601	Data Management and Analytics for IoT	4	0	4	20	30	50	-	100
5	VII	25EE10MP0701	Minor Project	0	8	4	50	-	-	50	100
TOTAL				**14**	**8**	**18**					

Multidisciplinary Minor (MDM) Track-1): IoT for Environmental Sustainability

Sr. No.	Semester	Course Code	Course Name	Offered To (Name of the Department)
1	III	24EE01TH0305-1	Introduction to IoT system Design	All - CSE, EC, EE, BME
2	IV	24EE01TH0405-1	Programming for Environmental IoT	All -CSE, EC, EE, BME
3	V	24EE01TH0505-1	IoT Privacy and Security	All -CSE, EC, EE, BME
4	VI	24EE01TH0605-1	Use cases of Environmental IoT	All -CSE, EC, EE, BME

Multidisciplinary Minor (MDM) Track-2: Integrated Circuit Design (IC design)

Sr. No.	Semester	Course Code	Course Name	Offered To (Name of the Department)
1	III	24EE01TH0305-2	Basics of Chip Design using Verilog HDL	All - CSE, EC, EE, BME
2	IV	24EE01TH0405-2	MIPS Processor Design and Testing	All -CSE, EC, EE, BME
3	V	24EE01TH0505-2	Chip Verification using System Verilog	All -CSE, EC, EE, BME
4	VI	24EE01TH0605-2	VLSI Physical Design	All -CSE, EC, EE, BME

Course Code	25EE01TP0101			
Category	Engineering Science Course			
Course Title	Basic Electronics and Computer Fundamentals			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	I

Course Outcomes

On successful completion of the course, students will be able to

1. Apply the knowledge of basic laws to analyze simple DC circuits.
2. Design and analyze simple diode and MOSFET circuits.
3. Apply foundational knowledge to analyze and compare computer system based on their specifications and operations.
4. Explain the characteristics of network topologies and describe the functions and structure of the OSI and TCP/IP models.

Syllabus

Module I:

Basic circuit elements and RLC Circuit: circuit elements resistor, inductor and capacitor, Ohm's Law and Kirchhoff's Laws; Analysis of series, parallel circuits excited by independent voltage sources; energy sources, dependent sources, star- delta transformation.

Module II:

Semiconductors and p-n junction diode: Intrinsic and extrinsic semiconductors, Dependence of Fermi level on carrier-concentration and temperature, Current flow in semiconductors, diffusion and drift, p-n junction diode, forward bias, reverse bias, Application of diode, Introduction to Bipolar Junction Transistors

Module III

MOSFET Transistors and Biasing: MOSFET construction and working principle, VI characteristics, AC/DC load line concept, Operating Point Analysis, need of biasing, biasing techniques, bias stabilization, compensation techniques, Application of MOSFET as Amplifier.

Module IV

Computer Basics: Introduction to Computer: Generation of Computer, Computer Architecture: Input /Output devices, CPU, Memory: Primary and secondary, Basic data encoding, Software: System, Application, Driver, Ports in Computer: Display, Ethernet, USB.

Module V

Network Fundamentals: Network Topologies, Types of networks: LAN, WAN, MAN, Network Standards ,OSI and TCP/IP models.

Textbooks

1. D. P. Kothari, I J Nagrath, Basic Electrical and Electronics Engineering, Second Edition, Mc Graw Hill,2020.
2. Donald Neamen “Electronic Circuits: Analysis and Design” Third Edition, McGraw-Hill Publication
3. Brookshear J. G., “Computer science: an overview”, Eleventh Edition, Addison-Wesley Publishing Company; 2011.

Re**ference Books**

1. E Balagurusamy, Fundamentals of Computers, Tata Mcgraw Hill Publications
2. Adel S. Sedra, Kenneth C. Smith, Arun N. Chandorkar: Microelectronic Circuits: Theory and Applications: Seventh Edition, Oxford University Press, 2017.
3. S. K. Bhattacharya, " Basic Electrical and Electronics Engineering", Pearson Education India Publications.

LIST OF EXPERIMENTS:

1. To measure voltage and current across a resistor and validate Ohm's Law.
2. To Verify Kirchhoff's Voltage and Current Laws using series and parallel circuit configurations.
3. To Convert a given resistive network from star to delta and vice versa and analyze the circuit behavior.
4. To Determine resonance frequency in a series RLC circuit and observe voltage variations.
5. Study the forward and reverse bias characteristics of a diode and determine threshold voltage.
6. To design and observe output waveforms of rectifier circuits using a diode.
7. To Plot VI characteristics of an n-channel MOSFET and analyze its operating regions.
8. To design and test a MOSFET-based amplifier circuit and observe its gain.
9. Identify and learn the functionality of different computer components like CPU, RAM, and storage devices.
10. To set up and test different network topologies (star, bus, ring) and analyze data transmission performance using Packet Tracer.

Course Code	25HS03TH0103			
Category	Basic Science Course			
Course Title	Probability and statistics			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	I

Course Pre-requisite : Basics of Probability and Statistics**.**

Course Objective:

The objective of this course is to expose student to understand the basic importance fundamental principles of probability, including probability distributions, random variables, basic statistical methods used for data analysis, inferential statistics, hypothesis testing, confidence intervals, and regression analysis in computer science and Information technology.

Course Outcomes

On successful completion of the course, student shall be able to

1. Identify and differentiate between discrete and continuous random variables, and interpret probabilities obtained from standard probability distributions.
2. To analyze and interpret stochastic models, including calculating probabilities, transition probabilities, and steady-state probabilities within stochastic systems.
3. Grasp the fundamental concepts of curve fitting like regression techniques, model selection, and the use of different types of curves or functions to approximate data.
4. Understand the fundamental concept of hypothesis testing, , significance levels, p-values, and the basic logic behind hypothesis testing.
5. To apply MLE to various statistical models, such as linear regression, exponential distribution, etc.

Syllabus

Module 1 (8 hours)

Measure of central tendency, quartile, inter quartile range and outliers**,** Probability spaces, conditional probability, independence, Discrete random variables, Continuous random variables , Expectation and variances, Binomial distribution, Poisson distribution, Normal distribution and their applications.

Module 2: (8 Lectures)

Joint probability function, Introduction to stochastic process, random walk, stationary and auto regressive process, transition probability Matrix, Discrete time Markov chain and its applications in queueing problems.

Module 3: (8 hours)

Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas and more general curves, correlation and regression – Rank correlation, Multiple regression and correlation and its application in analysis of data.

Module 4: (8 Lectures)

Sampling Distributions, Point and Interval Estimations, Testing of Hypothesis for single mean and proportion.

Module 5: (7 Lectures):

Testing of Hypothesis for difference of mean and proportion, Test for ratio of variances - Chi-square test for goodness of fit and independence of attributes, maximum likelihood estimation

Text Books:

1. M R. Spiegal , Theory and Problems of probability and statistics :,2nd ed :,Schaum series
2. S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.

Reference Books:

1. Maurtis Kaptein, Statistics for data science, An introduction to probability, statistics and Data Analysis, Springer 2022.
2. Jay L Devore, Probability and Statistics for Engineering and sciences, 8th edition, Cenage learning.

Course Code	25EE01TP0102			
Category	Engineering Science Course			
Course Title	Digital Logic Design			
Scheme & Credits	L	P	Credits	Semes ter
	3	2	4	I

Course Outcomes

Upon the completion of this course, students will demonstrate the ability to:

1. Understand Number Systems and its conversions
2. Apply various optimization techniques to minimize digital circuits.
3. Design combinational logic circuits.
4. Analyze and design asynchronous and synchronous sequential circuits.

Syllabus

Module-I

Basics of Digital Electronics: Motivation for digital systems: Number Systems and arithmetic's, Representation of Signed Numbers, Boolean algebra, Logic gates, SOP, POS, Minimization of Switching functions using Karnaugh-maps.

Module-II

Timing issues in Digital Circuit: Fan-In, Fan-Out, Propagation Delay, Power Dissipation, Noise Margin

Module-III

Combinational Circuit Design: Adders, Subtractors, Multiplexer,,De-multiplexers, Encoders, Decoders, Code Converters, Comparators.

Module-IV

Sequential Circuit Design-I: Storage elements, Flip-flops and latches: D, T, JK, SR flip-flops: level triggered, edge triggered, Master Slave flip-flop, flip flop conversion, timing analysis.

Module-V

Sequential circuit Design-II: Design of asynchronous and synchronous counters, Registers & Shift registers, Application of shift register: Ring counter, Johnson counter.

Module-VI

Design of synchronous sequential circuit using Mealy model and Moore model

Textbooks:

1. D.V.Hall,"DigitalCircuitsandSystems",TataMcGrawHill,1989.
2. Modern Digital Electronics: R. P Jain ,TataMcGrawHill,3rd Edition.

Reference Books:

1. Digital Logic and Computer Design: Morris Mano, PHI,3rd Edition.

LIST OF EXPERIMENTS:

1. To verify truth table of different logic gates.
2. Design basic logic gates using universal gate and verify its truth table.
3. To verify following Boolean expressions.

   Y=A+AB+A'B'

4. To implement the following arithmetic circuits using logic gates IC's
   a) Half adder
   b) Full subtractor
5. Implement the function F = Σm (1,3,5,7,8,9,11,13,15) +d (12,14) using 16:1 and 8:1 multiplexer.
6. Verify the truth table of SR, JK, T and D flip flop.
7. To study the following functions of Shift register.
   a) SIPO
   b) PIPO
   c) PISO
   d) SISO
8. Design and verify 2-bit synchronous down counter using S-R flip-flop.
9. Design and verify the functionality of a sequence detector to detect the sequence 101 using Mealy and Moore model and use J-K flop-flop to implement the design.

Course Code	25EE01TP0103			
Category	Program Core Course			
Course Title	Python Programming			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	I

Pre-requisite Knowledge:

The following foundational knowledge is expected:

- **Basic Computer Literacy:** Familiarity with operating a computer, using the internet, and basic software applications.
- **Basic Programming Concepts (Desirable):** A conceptual understanding of programming concepts like variables, data types, and control flow would be beneficial but is not strictly required.
- **English Language Proficiency:** Ability to understand and articulate concepts in English, as Python documentation and programming resources are predominantly in English.

Course Description:

This course provides a comprehensive introduction to Python programming, covering fundamental concepts and practical applications. Learners will begin with the basics of Python syntax, data types, operators, and control flow. The course progresses to object-oriented programming (OOP) principles, error and exception handling, file handling, and advanced Python programming concepts like regular expressions, modules, and libraries. The course also covers graphical user interface (GUI) development, web programming with CGI, and various Python applications, including networking, data processing, and database connectivity. The course integrates hands-on lab exercises to reinforce learning and develop practical programming skills.

Course Objectives:

Upon successful completion of this course, learners will be able to:

- Understand the core concepts of Python programming.
- Write and execute Python programs.
- Work with Python's data types, variables, operators, and control flow statements.
- Apply object-oriented programming (OOP) principles in Python.
- Implement error handling and exception handling in Python.
- Perform file input and output operations in Python.
- Use regular expressions for pattern matching and text processing.
- Create and use Python modules and libraries.
- Develop basic GUI applications.
- Build simple web applications using CGI.
- Apply Python for networking, data processing, and database applications.

Syllabus:

Unit 1: Introduction to Python Programming

1.1: Python Fundamentals

- 1.1.1: Introduction to Python
  - 1.1.1.1: What is Python?
  - 1.1.1.2: Python's Popularity and Use Cases
- 1.1.2: Python Syntax and Structure
  - 1.1.2.1: Python's Indentation and Block Structure
  - 1.1.2.2: Writing Your First Python Program
- 1.1.3: Writing and Executing Python Programs
  - 1.1.3.1: Using Python Interactive Mode (REPL)
  - 1.1.3.2: Running Python Scripts from Command Line

1.2: Python Basics

- 1.2.1: Variables and Data Types
  - 1.2.1.1: Understanding Variables and Naming Conventions
  - 1.2.1.2: Numeric Data Types (int, float)
  - 1.2.1.3: Text Data Type (str)
- 1.2.2: Operators and Expressions
  - 1.2.2.1: Arithmetic Operators
  - 1.2.2.2: Comparison Operators
  - 1.2.2.3: Logical Operators
- 1.2.3: Input and Output Operations
  - 1.2.3.1: Using input() for User Input
  - 1.2.3.2: Printing Output with print()

Lab Exercises -

Hello, World! Program

Write a Python program that prints "Hello, World!" to the console.

Interactive Mode Basics

Use Python's interactive mode to perform basic arithmetic operations like addition, subtraction, multiplication, and division.

User Input and Display

Create a Python script that takes user input for their name and displays a personalized greeting.

Calculate Rectangle Area

Write a Python program that calculates and prints the area of a rectangle. Prompt the user for the length and width.

Temperature Conversion

Create a Python script that converts a temperature from Fahrenheit to Celsius. Prompt the user for the temperature in Fahrenheit and display the result in Celsius.

Unit 2: Control Flow and Loops

2.1: Control Statements

- 2.1.1: Conditional Statements (if, elif, else)
  - 2.1.1.1: Simple if Statements
  - 2.1.1.2: elif for Multiple Conditions
- 2.1.2: Logical Operators and Conditions
  - 2.1.2.1: Using Logical AND, OR, NOT
  - 2.1.2.2: Complex Conditions
- 2.1.3: Switch-Case (if-elif-else)
  - 2.1.3.1: Implementing Switch-Like Behavior

2.2: Loops

- 2.2.1: While Loops
  - 2.2.1.1: Using while Loops for Iteration
  - 2.2.1.2: Controlling Loops with break and continue
- 2.2.2: For Loops
  - 2.2.2.1: Iterating Over Sequences (Lists, Strings)
  - 2.2.2.2: Using range() for Controlled Iteration
- 2.2.3: Loop Control Statements (break, continue)
  - 2.2.3.1: Breaking Out of a Loop
  - 2.2.3.2: Skipping Iterations with continue

Lab Exercises -

Even or Odd Checker

Implement a Python program that checks if a given number is even or odd and prints the result.

Largest among Three Numbers

Write a Python script that finds and prints the largest among three numbers entered by the user.

Factorial Calculation with a While Loop

Create a Python program that calculates and prints the factorial of a number entered by the user using a while loop.

Fibonacci Sequence with For Loop

Use a for loop to generate and print the Fibonacci sequence up to a specified number of terms. Prompt the user for the number of terms.

Sum of Prime Numbers

Write a Python program that calculates and prints the sum of all prime numbers in a given range. Prompt the user for the range.

Unit 3: Object-Oriented Programming (OOP) in Python

3.1: OOP Principles

- 3.1.1: Classes and Objects
  - 3.1.1.1: Defining Classes and Objects
  - 3.1.1.2: Constructors and Instance Variables
- 3.1.2: Inheritance and Polymorphism
  - 3.1.2.1: Creating Subclasses
  - 3.1.2.2: Overriding Methods
- 3.1.3: Encapsulation and Abstraction
  - 3.1.3.1: Access Modifiers (public, private, protected)
  - 3.1.3.2: Achieving Abstraction through Interfaces

3.2: Advanced OOP Concepts

- 3.2.1: Constructors and Destructors
  - 3.2.1.1: Parameterized Constructors
  - 3.2.1.2: Destructor in Python
- 3.2.2: Method Overloading and Overriding
  - 3.2.2.1: Overloading Methods
  - 3.2.2.2: Overriding Methods with super()
- 3.2.3: Class Variables and Instance Variables
  - 3.2.3.1: Understanding Class Variables
  - 3.2.3.2: Using Instance Variables

Lab Exercises -

Simple Calculator Class

Define a Python class for a simple calculator that has methods for addition and subtraction. Allow the user to perform calculations using objects of this class.

Class Inheritance Hierarchy

Create a class hierarchy with a base class and two derived classes. Demonstrate inheritance by accessing attributes and methods of each class.

Method Overriding

Implement method overriding in a Python class. Create a base class with a method and override it in a derived class.

Encapsulation Demonstration

Use encapsulation to restrict access to class attributes. Create a class with private attributes and demonstrate encapsulation principles.

Abstract Geometric Shape Class

Create an abstract class representing a geometric shape with abstract methods like area and perimeter. Define derived classes (e.g., Circle, Rectangle) to implement these methods.

Unit 4: Error Handling and Exception Handling

4.1: Exception Handling

- 4.1.1: Introduction to Exceptions
  - 4.1.1.1: Understanding Exceptions in Python
  - 4.1.1.2: Common Built-in Exceptions
- 4.1.2: Handling Exceptions with try and except
  - 4.1.2.1: Using try-except Blocks
  - 4.1.2.2: Handling Multiple Exceptions
- 4.1.3: Custom Exceptions
  - 4.1.3.1: Creating Custom Exception Classes
  - 4.1.3.2: Raising Exceptions

4.2: File Handling (I/O)

- 4.2.1: Reading and Writing Files
  - 4.2.1.1: Opening and Closing Files
  - 4.2.1.2: Reading and Writing Text Files
- 4.2.2: Working with Text and Binary Files
  - 4.2.2.1: Reading and Writing Binary Files
  - 4.2.2.2: Text Encoding and Decoding
- 4.2.3: File Handling Best Practices
  - 4.2.3.1: Using 'with' Statements
  - 4.2.3.2: Error Handling in File Operations

Lab Exercises -

Custom Exception Handling

Write a Python program that raises and handles a custom exception. Define a custom exception class and demonstrate its usage.

File Exception Handling

Implement a function that reads data from a file and handles file-related exceptions such as FileNotFoundError and PermissionError.

Division by Zero Handling

Create a Python program that simulates division by zero and handles the ZeroDivisionError exception gracefully.

File Handling with Multiple Exceptions

Modify a file reading program to handle both FileNotFoundError and PermissionError exceptions.

Finally Block Usage

Develop a program that demonstrates the use of the finally block in exception handling. Open a file and ensure it is properly closed even if exceptions occur.

Unit 5: Advanced Python Programming

5.1: Regular Expressions

5.1.1: Introduction to Regular Expressions
5.1.1.1: What are Regular Expressions?
5.1.1.2: Use Cases for Regular Expressions
5.1.2: Pattern Matching and Text Processing
5.1.2.1: Matching Patterns with re Module
5.1.2.2: Extracting Data from Text
5.1.3: Regex in Python
5.1.3.1: Using Regular Expressions in Python
5.1.3.2: Regex Functions and Methods

5.2: Modules and Libraries

5.2.1: Creating and Using Modules
5.2.1.1: Writing Your Own Modules
5.2.1.2: Importing Modules
5.2.2: Standard Library Modules
5.2.2.1: Exploring Built-in Modules (math, datetime)
5.2.2.2: Working with OS and sys Modules
5.2.3: Third-party Libraries and Packages
5.2.3.1: Using pip for Package Installation
5.2.3.2: Popular Third-party Libraries (requests, pandas)

Lab Exercises -

Email Validation with Regular Expressions

Write a Python program that validates email addresses using regular expressions. Prompt the user for an email address and validate it.

Phone Number Extraction

Create a Python script that extracts phone numbers from a given text using regular expressions.

Custom Python Module

Develop a Python module with functions to perform basic arithmetic operations. Import this module into another script and use its functions.

Exploring Built-in Modules

Explore Python's built-in modules like math and datetime. Use them to perform mathematical operations and work with date and time.

Third-Party Library Usage

Install and use a third-party library (e.g., requests) to fetch data from a web API. Retrieve data and display it in your Python script.

Unit 6: Graphical User Interfaces (GUI) and Web Programming

6.1: GUI Development

6.1.1: Introduction to GUI Programming
6.1.1.1: GUI vs. Command Line Interfaces
6.1.1.2: GUI Frameworks in Python
6.1.2: Widgets and Event Handling
6.1.2.1: Creating Widgets (Buttons, Labels)

6.1.2.2: Handling User Events (Clicks, Input)

6.2: Web Programming with CGI

6.2.1: Introduction to CGI

6.2.1.1: What is CGI and Its Purpose

6.2.1.2: CGI in Web Development

6.2.2: Handling HTTP Requests

6.2.2.1: Receiving and Processing Requests

6.2.2.2: Generating HTTP Responses

6.2.3: Building Interactive Web Applications

6.2.3.1: Form Handling with CGI

6.2.3.2: Implementing Data Processing

Lab Exercises -

Simple tkinter GUI

Design a simple tkinter GUI application with buttons and labels. Implement functionality to update labels when buttons are clicked.

GUI Button Actions

Create a Python program that responds to user button clicks in a tkinter GUI. Perform actions like displaying messages when buttons are clicked.

HTML Form Handling

Build an HTML web form with input fields and a submit button. Create a Python CGI script to handle form submissions and display the entered data.

Form Data Validation

Extend the previous exercise to validate form data in the Python CGI script. Check for required fields and display validation messages.

Dynamic Web Content with CGI

Implement a Python CGI script that generates dynamic web content based on user requests. Create a simple web application that displays different content based on URL parameters.

Unit 7: Python Applications

7.1: Networking and Serialization

7.1.1: Networking Basics in Python

7.1.1.1: Introduction to Networking Protocols

7.1.1.2: Creating Client and Server Sockets

7.1.2: Socket Programming

7.1.2.1: Building Networked Applications

7.1.2.2: Data Transfer and Communication

7.1.3: Serialization (JSON and Pickle)

7.1.3.1: Serialization Overview

7.1.3.2: JSON and Pickle for Data Serialization

7.2: Data Processing and Analysis

7.2.1: Introduction to NumPy and Pandas

7.2.1.1: What is NumPy?

7.2.1.2: Overview of Pandas

7.2.2: Data Manipulation with NumPy

7.2.2.1: Creating NumPy Arrays

7.2.2.2: Array Operations and Manipulation

7.2.3: Data Analysis with Pandas

7.2.3.1: Working with DataFrames

7.2.3.2: Data Cleaning and Exploration

7.3: Database Applications

7.3.1: Database Connectivity in Python

7.3.1.1: Database Management Systems (DBMS)

7.3.1.2: Connecting to Databases

7.3.2: SQL Queries and Database Operations

7.3.2.1: Structured Query Language (SQL)

7.3.2.2: Executing SQL Queries from Python

7.3.3: Building Database-driven Applications

7.3.3.1: Integrating Python with Databases

7.3.3.2: CRUD Operations in Database Applications

Lab Exercises -

Client-Server Communication

Create a simple client-server program using socket programming in Python. Implement basic communication between the client and server.

Enhanced Client-Server Communication

Extend the client-server program to support data transfer between the client and server. Implement sending and receiving data between the two.

JSON Serialization and Deserialization

Serialize Python objects into JSON format and then deserialize them back into Python objects. Demonstrate data interchange between JSON and Python.

Pickle Serialization

Use Pickle to serialize and deserialize Python objects. Save Python objects to a file using Pickle and then load them back.

Networked Data Exchange

Develop a Python script that simulates data exchange over a network using sockets and serialization. Send and receive data between client and server.

Basic NumPy Operations

Create a NumPy array and perform basic operations like addition, subtraction, multiplication, and division.

Data Filtering and Selection

Implement data filtering and selection using NumPy arrays. Filter data based on specific conditions and criteria.

Data Analysis with Pandas

Use Pandas to read data from a CSV file and perform data analysis. Calculate statistics like mean, median, and standard deviation on the dataset.

DataFrame Manipulation

Explore DataFrame manipulation in Pandas. Sort, filter, and perform various operations on a dataset loaded into a DataFrame.

Database CRUD Operations

Connect to a SQLite database in Python and perform CRUD (Create, Read, Update, Delete) operations using SQL queries. Create a Python script that interacts with a database.

Course Summary:

This course provides a comprehensive introduction to Python programming, covering fundamental concepts and practical applications. The course covers Python basics, control flow, object-oriented programming, error handling, file handling, advanced Python programming, GUI development, web programming with CGI, and Python applications in networking, data processing, and databases. The course is structured to provide a balance of theoretical knowledge and practical skills through hands-on lab exercises. By the end of the course, learners will be equipped with the Python programming skills necessary to develop a variety of applications.

Course Flow of Learning:

The course is structured into seven units, designed to build a progressive understanding of Python programming:

- **Unit 1: Introduction to Python Programming:** This unit introduces Python fundamentals, syntax, and basic input/output operations.
- **Unit 2: Control Flow and Loops:** This unit covers conditional statements and loop structures for controlling program execution.
- **Unit 3: Object-Oriented Programming (OOP) in Python:** This unit introduces the principles of object-oriented programming and their implementation in Python.
- **Unit 4: Error Handling and Exception Handling:** This unit covers handling exceptions and file input/output operations.
- **Unit 5: Advanced Python Programming:** This unit introduces regular expressions, modules, and libraries.
- **Unit 6: Graphical User Interfaces (GUI) and Web Programming:** This unit covers GUI development and web programming with CGI.
- **Unit 7: Python Applications:** This unit covers Python applications in networking, data processing, and database connectivity.

Course Outcomes:

Upon successful completion of this course, learners will be able to:

- Write Python programs to solve basic programming problems.
- Apply Python data structures and control flow statements.
- Develop object-oriented programs in Python.
- Handle errors and exceptions in Python programs.
- Perform file handling operations for data storage and retrieval.
- Use regular expressions for text manipulation.
- Create and use Python modules and libraries.
- Develop simple GUI and web applications.
- Apply Python for networking, data processing, and database connectivity.
- Design and implement Python-based applications.

Skills Gained:

Upon completion of this course, learners will gain the following skills:

- **Python Programming Fundamentals:** Writing and executing Python code, understanding syntax, data types, operators, and control flow.
- **Object-Oriented Programming (OOP):** Designing and implementing classes, objects, inheritance, and polymorphism.
- **Exception Handling:** Handling errors and exceptions in Python programs.
- **File Handling:** Reading from and writing to files in Python.
- **Regular Expressions:** Using regular expressions for pattern matching and text processing.
- **Modules and Libraries:** Creating and using Python modules and libraries.
- **GUI Development:** Developing basic graphical user interfaces.
- **Web Programming (CGI):** Building simple web applications with CGI.
- **Networking:** Creating client and server applications using sockets.
- **Data Processing:** Using NumPy and Pandas for data manipulation and analysis.
- **Database Connectivity:** Connecting to and interacting with databases using Python.
- **Problem-Solving:** Applying Python programming to solve real-world problems.
- **Application Development:** Building Python-based applications.

Course Modules:

Module 1: Python Fundamentals and Control Flow

- Unit 1: Introduction to Python Programming
- Unit 2: Control Flow and Loops

Module 2: Object-Oriented Programming and Error Handling

- Unit 3: Object-Oriented Programming (OOP) in Python
- Unit 4: Error Handling and Exception Handling

Module 3: Advanced Python and Applications

- Unit 5: Advanced Python Programming
- Unit 6: Graphical User Interfaces (GUI) and Web Programming
- Unit 7: Python Applications

Module 1: Python Fundamentals and Control Flow
Module 2: Object-Oriented Programming and Error Handling
Module 3: Advanced Python and Applications

Course Code	25EE01TP0104			
Category	Co-Curricular Activity			
Course Title	Design Thinking and Innovation			
Scheme & Credits	L	P	Credits	Semester
	1	2	2	I

Pre-requisite Knowledge:
A basic understanding of problem-solving, general awareness of business or societal challenges, and a willingness to learn and collaborate. No specific technical or design background is strictly necessary, but an open and curious mindset is crucial. Familiarity with common Indian contexts and user behaviors will be beneficial for case studies and application exercises.

Course Description:
This course provides a comprehensive introduction to Design Thinking, a human-centered and iterative problem-solving approach. Learners will explore the fundamental principles, key phases, and practical techniques of Design Thinking. Through case studies, discussions, and hands-on activities, participants will develop the skills to identify user needs, define problems effectively, generate innovative ideas, and create and test prototypes. This course equips learners with a valuable framework for driving innovation and addressing complex challenges across various industries.

Course Objectives:
Upon completion of this course, learners will be able to:

- Understand the core concepts, principles, and benefits of Design Thinking.
- Identify and define problems effectively using various Design Thinking techniques.
- Develop empathy for users and stakeholders to uncover their needs and pain points.
- Generate a wide range of creative ideas through diverse ideation methods.
- Evaluate and prioritize ideas based on feasibility and impact.
- Create low-fidelity prototypes to visualize and test potential solutions.
- Apply different testing methods to gather user feedback and iterate on designs.
- Communicate ideas effectively through storytelling and presentations.
- Recognize the applications of Design Thinking in various contexts.

Course Summary:
This course begins by establishing the foundational understanding of Design Thinking, its importance, and its various stages. It then delves into the crucial initial phases of defining the problem, emphasizing empathy and stakeholder analysis. The course progresses to ideation, where learners explore numerous techniques for generating creative solutions, followed by methods for evaluating and prioritizing these ideas. Finally, it covers the essential stages of prototyping and testing, highlighting the iterative nature of Design Thinking and the importance of user feedback. Throughout the course, practical application through case studies and exercises is emphasized.

Course Flow of Learning:

1. **Introduction to Design Thinking:** Understanding the what, why, and how of Design Thinking, its mindset, and key principles.
2. **Problem Definition:** Learning to identify and clearly articulate problems by understanding user needs and stakeholder perspectives.
3. **Empathizing and Identifying Real Problems:** Deep diving into user observation and employing

empathetic design techniques to uncover underlying issues.

4. **Ideation:** Exploring a range of creative techniques to generate a multitude of potential solutions.
5. **Prioritization:** Learning methods to evaluate and select the most promising ideas based on various criteria.
6. **Prototyping and Testing:** Creating models and MVPs to test ideas with users and gather feedback for iteration.
7. **Idea Pitching:** Developing effective storytelling and presentation skills to communicate design concepts.

Course Outcomes:

Upon successful completion of this course, learners will be able to:

- Apply the Design Thinking framework to solve real-world problems.
- Conduct effective user research and develop empathy maps and personas.
- Formulate clear and concise problem statements.
- Facilitate and participate in brainstorming and ideation sessions.
- Utilize various techniques for evaluating and prioritizing ideas.
- Create basic prototypes to test design concepts.
- Gather and analyze user feedback to refine solutions.
- Present design ideas persuasively.

Skills Gained:

- Problem Identification and Definition
- Empathy Development
- User Research
- Stakeholder Analysis
- Brainstorming and Ideation
- Creative Thinking
- Idea Evaluation and Prioritization
- Prototyping (basic)
- Testing and Feedback Gathering
- Communication and Presentation
- Collaboration

Course Modules:

Module 1: Understanding the Foundation of Design Thinking

- Unit 1: Introduction and Background of Design Thinking
- Unit 2: Getting started with Problem Statement (Initial stages of understanding the problem)

Module 2: Deep Dive into Problem Understanding and Ideation

- Unit 2: Getting started with Problem Statement (In-depth problem analysis)
- Unit 3: Identifying real problem
- Unit 4: Deliver on Big Idea: Ideation and Prioritization

Module 3: Bringing Ideas to Life and Validation

- Unit 5: Model creation and Idea pitching

Syllabus:

Unit 1. Introduction and Background of Design Thinking

Introduction, Knowing the team and course, The strategy of Innovation in design thinking. Why is Design Thinking required? Industries in Design Thinking, what is Design Thinking? Design thinking is a way of thinking. Design Thinking Tips and Anecdotes, Design Thinking Mindset Design thinking vs Scientific approach, Analysis vs. Synthesis in design thinking, Divergent Thinking vs. Convergent Thinking in design thinking, key phases of the design thinking process, Case Studies on Design thinking, Fundamentals of Design Thinking, Stages of Design Thinking, Design Thinking Skills, The process of Design Thinking, Design thinking framework, Why Design Thinking Works, Incorporating design thinking into your work, Limitations of a design thinking process, Advantages of the design thinking approach, Examples of design thinking success, Planning a Design Thinking Project, Benefits of the design thinking approach, Applications of design thinking.

Unit 2. Getting started with Problem Statement

Introduction What is a Problem Statement? Initial questions, Case Studies, Problem Clarification, Role of the Stakeholders, Activities for Problem Clarification, Point-Of-View (POV) in Design Thinking, Empathy Map, Understand and Define Problem Statement, Problem Analysis, Root Cause analysis tools, Defining Metrics, Persona Identification, User Personas, Stakeholder Map, who are Stakeholders? What is their role? Stakeholders of Few Companies, Creating a Stakeholder Map, Example Scenarios, Dos and Don'ts During Problem statement identification, Reformulation of the Problem.

Unit 3. Identifying real problem

Introduction, what is a Design Thinking Problem Statement? steps to create a design thinking problem statement, Learn Why and How to Focus on User Problem, Observation Phase, The Power to Observe, Useful Instrument for Observation Sessions, Tips for Observing, Practice problem Identification using Empathetic Design, Methods for Empathetic Design, Inquiry Vs. Observation, Learn As -is-state, Practice problem Identification using As-is-state long answer, Point-of-View phase for defining a problem, Characterizing the target group, Top of Form, Description of customer needs.

Unit 4. Deliver on Big Idea: Ideation and Prioritization

Introduction, Ideation Phase, The Ideation Funnel, Divergent and Convergent Thinking, Techniques for Clearing the Mind, The creative process and creative principles, Understanding Creativity, Creative Principles, Creativity Techniques, Brainstorming, Example of a Brainstorming Session, Mind Mapping, SCAMPER, Random Word Association, Rapid Prototyping / Design Sprint, Learn various Idea Generation Techniques, Creativity and Idea Generation, Common Techniques, Reverse Brainstorming, Role Playing, Analogies, Real-life Examples, Forced Connection, Visual Thinking, Storytelling, Practice Ideation, Overcoming Creative Blocks, Unrealistic Expectations, The Right Way to Take Risks, Examples of Creative Thinking, Evaluation of Ideas, Criteria / Metrics for Evaluation, Testing the Ideas, Refining and Selecting the Best Ideas, Transparency in the Evaluation Process, Pass/Fail Method, Idea Evaluation Matrix, SWOT Analysis, Criteria for Prioritization, Idea Prioritization Techniques -

a. Dot Voting
b. Impact vs. feasibility matrix
c. Weighted Scoring
d. Affinity Mapping
e. Cost-Benefit Analysis (CBA)
f. MoSCoW Method
g. Kano Model
h. Eisenhower Matrix

Implementing Prioritization, Re-Prioritization, Learn To-be Scenario, To-Be Scenario Map, use a to-do list to plan, How to Write an effective to-do List, Using Technology for to-do lists, Strategies for practicing prioritization, Urgent vs. Important Matrix, ABC Analysis, Value-based prioritization, Time-based prioritization, The Pomodoro Technique, Time-Value Prioritization Funnel, Pareto Analysis, Time-blocking, Tips for maintaining focus and staying motivated when prioritizing tasks, How to adjust priorities as circumstances change.

Unit 5. Model creation and Idea pitching

Introduction, Prototyping Phase, Learn Wireframe and Model creation, Introduction to Minimum Viable Product, Benefits Of MVP Development, What is expected from Minimum Viable Product, Practice Wireframe creation, End to end connecting from Problem to MVP, Testing Phase, Types of Testing, White and Black Box Testing, Unit Testing, Integration testing, Unit Testing Vs Integration Testing, System Testing, System Testing Vs Functional Testing, Acceptance Testing, User Acceptance Testing (UAT), Operational Acceptance Testing (OAT), Contract Acceptance Testing, Alpha/Beta Testing, Performance Testing, Security Testing, Usability Testing, Testing Techniques, Test Cases and Test Scripts, Bug Tracking and Reporting, User Feedback and Testing, Continuous Testing, Challenges and Best Practices, Tips for Interviews and Surveys, Kano Model, Desirability testing, Storytelling and Idea Presentation.

Reference Books:

1. "Design Thinking: Integrating Innovation, Customer Experience, and Brand Value" by Thomas Lockwood
2. "The Design Thinking Playbook: Mindful Digital Transformation of Teams, Products, Services, Businesses and Ecosystems" by Michael Lewrick, Patrick Link, Larry Leifer
3. "Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days" by Jake Knapp, John Zeratsky, Braden Kowitz
4. "Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation" by Tim Brown
5. "Creative Confidence: Unleashing the Creative Potential Within Us All" by Tom Kelley, David Kelley
6. "The Art of Innovation: Lessons in Creativity from IDEO, America's Leading Design Firm" by Tom Kelley
7. "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses" by Eric Ries

Course Code	25EE01PR0105			
Category	Vocational and Skill Enhancement Course			
Course Title	Computer workshop			
Scheme & Credits	L	P	Credit s	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Acquire a working knowledge of Linux fundamentals and Linux distributions.
2. Apply knowledge to comprehend system configurations and Linux graphical interfaces.
3. Independently perform fundamental command line operations in Linux.
4. Effectively employ common Linux applications for specific tasks and functionalities

Practical's based on:

The Linux Foundation: Linux Philosophy and Concepts, Linux Basics and System Startup.

Graphical Interface, System Configuration from the Graphical Interface, Common Applications, Command Line Operations, Finding Linux Documentation.

Processes, File Operations, Text Editors, User Environment, Manipulating Text, Network Operations.

The Bash Shell and bash Scripting, Introduction, Features and Capabilities, Syntax, Constructs.

Printing, Local Security Principles, Understanding Linux Security, root Privileges, sudo, Process Isolation, Limiting Hardware Access and Keeping Systems Current, Working with Passwords, Securing the Boot Process and Hardware Resources.

Remote access and managing processes through remote login

Text books

1) Linux BIBLE, Christopher Negus, Tenth Edition, Wiley 2020.

2) Linux for Beginners: An Introduction to the Linux Operating System and Command Line, Jason Cannon, O'Reilly, 2014.

Course Title: Liberal / Performing Arts

Course Code	Course Name	Sem	Hours/ week	Credit s	Continuous Evaluation Max. marks
25HS02PR0102-02	Fundamentals of Indian Classical Dance: Bharatnatayam	I	2	1	50
25HS02PR0102-03	Fundamentals of Indian Classical Dance: Kathak	I	2	1	50
25HS02PR0102-04	Introduction to Digital Photography	I	2	1	50
25HS02PR0102-05	Introduction to Basic Japanese Language	I	2	1	50
25HS02PR0102-06	Art of Theatre	I	2	1	50
25HS02PR0102-07	Introduction to French Language	I	2	1	50
25HS02PR0102-08	Introduction to Spanish Language	I	2	1	50
25HS02PR0102-09	Art of Painting	I	2	1	50
25HS02PR0102-10	Art of Drawing	I	2	1	50
25HS02PR0102-11	Nature Camp	I	2	1	50
25HS02PR0102-12	Developing Self-awareness	I	2	1	50
25HS02PR0102-13	Art of Poetry	I	2	1	50
25HS02PR0102-14	Creative and content writing	I	2	1	50
25HS02PR0102-15	Science of life through Bhagwad Gita	I	2	1	50
25HS02PR0102-16	Sanskrit Sambhashan- Spoken Sanskrit	I	2	1	50
25HS02PR0102-17	Kirtan Kala	I	2	1	50
25HS02PR0102-18	Introduction to German Language and culture	I	2	1	50
25HS04PR0102-1	Adventure Sports	I	2	1	50
25HS04PR0102-2	Introduction to Défense Forces & Obstacle Training	I	2	1	50
25HS04PR0102-3	First Aid & Disaster Management	I	2	1	50
25HS04PR0102-4	Basic Nutritional Course	I	2	1	50
25HS04PR0102-5	Stress Management Through Yoga & Meditation	I	2	1	50

Course Code	25HS02PR0102-02			
Category	Co-Curricular Activity			
Course Title	Fundamentals of Indian Classical Dance: Bharatnatayam			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Understand the importance of dance and Bharatnataym as an Indian dance form
2. Develop skills to perform the dance form at its basic level.
3. Evaluate their strengths and interest to take bridge course to give Pratham (1st level formal exam of Bharatnatayam).

Syllabus

1.Orientation in Bharatnatayam
2.Tattu Adavu till 8, NaattaAdavu 4 Steps, PakkaAdavu 1 step, Metta Adavu 1 Step, Kuditta Metta Adavu 4 Steps
3. Practice sessions
4.Tatta Kuditta Adavu (Metta), Tatta Kuditta Adavu (Metta) 2 Steps, Tirmanam Adavu 3 Steps, Kattu Adav - 3 Steps, Kattu Adav - 3 Steps
5. Practice sessions
6. Tiramanam (front) 3 Steps, Repeat of Tiramanam (Overhead) 3 Steps
7.Practice sessions
8.Final practice sessions and performances.

Recommended reading

1.Introduction to Bharata's Natyasastra, Adya Rangacharya, 2011
2.The Natyasastra and the Body in Performance: Essays on the Ancient Text, edited by Sreenath Nair, 2015
3.Bharatanatyam How to ... : A Step-by-step Approach to Learn the Classical Form, Eshwar Jayalakshmi, 2011

Course Code	25HS02PR0102-03			
Category	Co-Curricular Activity			
Course Title	Fundamentals of Indian Classical Dance: Kathak			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Understand the importance of dance and Kathak as an Indian dance form
2. Develop skills to perform the dance form at its basic level.
3. Evaluate their strengths and interest to take bridge course to give Prarambhik (1st level formal exam of Kathak).

Syllabus

1.Orientation in Kathak. Correct posture of kathak, Basic Movements and exercise Stepping, Chakkar of 5 count (Bhramari),

2.practice sessions of practical 1

3.Hastaks, Hastaks and Steppings, Reciting asamyukta Mudra shloka, Hastak and steppings

4.practice sessions of practical 3

5.Todas and Asamyukta hasta mudra shlok, Vandana of Shlok, 2 Todas and Vandana, Ghante Ki Tihai,

6.practice sessions of practical 5

7.2 1 Chakkardar Toda and Ginnti Ki Tihai, 2 Todas and 1 Chakkardar Toda, practice sessions

8.Final performances.

Recommended reading

1.Kathak Volume1 A "Theoretical & Practical Guide" (Kathak Dance Book), Marami Medhi & Debasish Talukdar, 2022, Anshika Publication (13 September 2022)

Course Code	25HS02PR0102-04			
Category	Co-Curricular Activity			
Course Title	Introduction to Digital Photography			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Develop an understanding of the technical aspects and aesthetics of Photography.
2. Apply the rules of digital photography for creating photographs.
3. Develop skills to enhance photographs through post processing.
4. Create a portfolio of their photographs in selected genre.

Syllabus

1. Orientation in digital photography**:** Genres, camera handling and settings
2. Rules of Composition
3. Rules of Composition: practice sessions
4. Understanding Exposure and Art of Pre-Visualization
5. Rules of Composition and Art of Pre-Visualization: practice sessions
6. Post Processing Photographs and Portfolio creation
7. Post Processing Photographs: practice sessions
8. Portfolio finalization and presentation in selected genre.

Reference material

1. Scott Kelby (2020) The Digital Photography Book: The Step-by-Step Secrets for how to Make Your Photos Look Like the Pros, Rocky Nook, USA
2. Larry Hall (2014) Digital Photography Guide: From Beginner to Intermediate: A Compilation of Important Information in Digital Photography, Speedy Publishing LLC, Newark
3. J Miotke(2010) BetterPhoto Basics: The Absolute Beginner's Guide to Taking Photos Like a Pro, AMPHOTO Books, Crown Publishing Group, USA

Course Code	25HS02PR0102-05			
Category	Co-Curricular Activity			
Course Title	Introduction to Basic Japanese Language			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to
1. Gain a brief understanding about Japan as a country and Japanese culture.
2. Develop ability to use vocabulary required for basic level communication in Japanese language.
3. Write and read the first script in Japanese language.
4. Frame simple sentences in Japanese in order to handle everyday conversations
5. Write in basic Japanese about the topics closely related to the learner.

Syllabus

1.Orientation about Japan, its language, and its culture
2.Communication Skills 1: Vocabulary for basic Japanese language
3. Practice sessions
4.Writing Skills 1: Reading and writing first script in Japanese
5.Practice sessions
6.Communication Skills 2: framing sentences
7.Practice sessions
8.Writing Skills 2: Write basic Japanese and practice

Recommended reading

1.Marugoto Starter (A1) Rikai - Course Book for Communicative Language Competences, by The Japan Foundation, Goyal Publishers & Distributors Pvt. Ltd (ISBN: 9788183078047)
2.Japanese Kana Script Practice Book – Vol. 1 Hiragana, by AmeyaPatki, Daiichi Japanese Language Solutions (ISBN: 9788194562900)

Course Code	25HS02PR0102-06			
Category	Co-Curricular Activity			
Course Title	Art of Theatre			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Understand and synthesize the working of the prominent genres of theatre across the world.
2. Apply the skill of voice and speech in theatre and public speaking
3. Apply the art of acting and also develop generic skills such as confidence, communication skills, self- responsibility, motivation, commitment, interpersonal skills, problem solving, and self-discipline.
4. Apply the skills, acquired related to technical/production aspects of theatre and also develop problem solving and interpersonal skills.

Syllabus

1. Orientation in theatre
2. Voice and Speech training
3. Voice and Speech training: practice sessions
4. Art of acting
5. Art of acting: practice sessions
6. Art of script writing
7. Art of script writing: practice sessions
8. Final performances

Reference books

1. Boleslavsky, R. (2022). Acting: The First Six Lessons (1st ed., pp. 1-92). Delhi Open Books.
2. Shakthi, C. (2017). No Drama Just Theatre(1st ed., pp. 1-171). Partridge.
3. Bruder, M., Cohn, L. M., Olnek, M., Pollack, N., Previto, R., &Zigler, S. (1986*).* A Practical Handbook for the Actor (1st ed.). Vinatge Books New York.

Course Code	25HS02PR0102-07			
Category	Co-Curricular Activity			
Course Title	Introduction to French Language			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Demonstrate basic knowledge about France, the culture and similarities/differences between India and France
2. Learn to use simple language structures in everyday communication.
3. Develop ability to write in basic French about themselves and others.
4. Develop ability to understand beginner level texts in French

Syllabus

List of Practicals

1. Orientation about France, the language, and culture
2. Communication Skills 1: Vocabulary building for everyday conversations
3. Practice sessions
4. Reading and writing Skills : Reading and writing simple text in French
5. Practice sessions
6. Communication Skills 2: listening comprehension
7. Practice sessions
8. Writing Skills: Write basic French and practice

Recommended reading

1. 15-minute French by Caroline Lemoine
2. Cours de Langue et de Civilisation Françaises by G. Mauger Vol. 1.1
3. Cosmopolite I by Natalie Hirschsprung, Tony Tricot

Course Code	25HS02PR0102-08			
Category	Co-Curricular Activity			
Course Title	Introduction to Spanish Language			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Demonstrate basic knowledge about Spain, the culture and similarities/differences between India and France
2. Learn to use simple language structures in everyday communication.
3. Develop ability to write in basic Spanish about themselves and others.
4. Develop ability to read and understand beginner level texts in Spanish

Syllabus

List of Practicals

1. Orientation about Spain, the language, and culture
2. Communication Skills 1: Vocabulary building for everyday conversations
3. Practice sessions
4. Reading and writing Skills: Reading and writing simple text in Spanish
6. Communication Skills 2: listening comprehension
7. Practice sessions
8.Writing Skills: Write basic Spanish and practice

Recommended reading

1. 15-Minute Spanish by Ana Bremon
2. Aula Internacional 1 by Jaime Corpas ,Eva Garcia, Agustin Garmendia.
3.Chicos Chicas Libro del Alumno by María Ángeles Palomino

Course Code	25HS02PR0102-09			
Category	Co-Curricular Activity			
Course Title	Art of Painting			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to
1.Become familiar with the basic methods, techniques & tools of painting.
2. Train the eye and hand to develop sense of balance, proportion and rhythm.
3. Develop the ability to observe and render simple natural forms.
4. Enjoy the challenging and nuanced process of painting.

Syllabus

1.Orientation in Painting tools & basics of lines, shapes, light, shadows and textures
2.The art of observation how to see shapes in drawing
3.Introduction Water colorhow to handle water paints
4.Introduction to acrylic colorshow to handle acrylic paints
5.Explore layering paint and capturing the quality of light with paint.
6. Create landscape painting
7.Create Abstract painting
8. Paint on Canvas (try to recreate any famous painting)

Reference material

1.Drawing made easy by Navneet Gala; 2015th edition
2.Alla Prima II Everything I Know about Painting--And More by Richard Schmid with Katie Swatland 3.Daily Painting: Paint Small and Often To Become a More Creative, Productive, and Successful Artist by Carol Marine

Course Code	25HS02PR0102-10			
Category	Co-Curricular Activity			
Course Title	Art of Drawing			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Become familiar with the basic methods, techniques & tools of drawing.
2. Train the eye and hand to develop sense of balance, proportion and rhythm.
3. Develop the ability to observe and render simple natural forms.
4. Enjoy the challenging and nuanced process of drawing.

Syllabus

1. Orientation in Drawing tools & basics of lines, shapes, light, shadows and textures
2. The art of observation how to see shapes in drawing
3. One/two-point basic linear perspective
4. Nature drawing and landscapes
5. Gestalt principles of visual composition
6. Figure drawing: structure and proportions of human body
7. Gesture drawing: expression and compositions of human figures
8. Memory drawing: an exercise to combine the techniques learnt

Reference material

1. Drawing made easy by Navneet Gala; 2015th edition
2. Perspective Made Easy (Dover Art Instruction) by Ernest R. Norling

Course Code	25HS02PR0102-11			
Category	Co-Curricular Activity			
Course Title	Nature Camp			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course, students will be able to

1. Develop an affinity with nature by observing and understanding it marvels with guidance from experts
2. Develop an understanding of the challenges and solutions associated with nature and its conservation.

Course content

In collaboration with the Forest Department and/or a local NGO working in the field of environment conservation, this course would be conducted in 24 hours. Students will be taken to a tiger reserve in Vidarbha region or Forest fringe villages or work with an NGO from Vidarbha region working on natural resource management. The camps (for 2 days) will cover any one of the following topics as decided by the course coordinator:

1. Awareness about each element of biodiversity (camps on moths, butterflies, birds, other wildlife etc)
2. Environment management (water, forest, wildlife) – practices of Forest Department in managing a tiger reserve, and other aspects of water and forest conservation.
3. Sustainable natural resource management - initiatives by rural communities and local NGOs
4. Man-animal conflict and solutions (socio-economic and technical) – role of local communities and Forest Department
5. Traditional practices in environment conservation – role of local communities and local NGOs

Course Code	25HS02PR0102-12			
Category	Co-Curricular Activity			
Course Title	Developing Self-awareness			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course objectives:
The course aims to develop students in their personal as well as professional life by means of graphotherapy, NLP, and Neurobics

Course Outcomes:
On completion of the course, students will be able to achieve the following:

CO1: Gain foundational understanding of graphology and through self-analysis will achieve greater awareness about their strengths and weaknesses & areas for personal growth
CO2: students will be equipped with tools and techniques for continuous self-improvement, using signature analysis and graphotherapy as part of their personal development journey
CO3: understand how to use Neuro Linguistic Programming (NLP) strategies to set and achieve goals effectively, overcoming mental blocks and limiting beliefs.
CO4: Enhance ability to absorb, retain, and recall information, which can benefit academic and professional performance.

Syllabus:

Practical 1: **The Power of Handwriting (Handwriting is Brainwriting)**
Practical 2: **Know yourself through handwriting**
Practical 3: **The Role of Signature in your life**
Practical 4: **Graphotherapy to enhance yourself in all ways**
Practical 5: **Neurolinguistic Programming , S.M.A.R.T Goal**
Practical 6: **Effective Communication Model, Rapport Building and Anchor**
Practical 7: **Brain Directives & Linguistic Presuppositions**
Practical 8: **Neurobics**

Course Code	25HS02PR0102-13			
Category	Co-Curricular Activity			
Course Title	Art of Poetry			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes:
To familiarize the students with the art of poetry and develop a sense of appreciation for the art

At the end of the course the student will be able to achieve the following:
CO1: Understand the origin and development of poetry
CO2: Appreciate the art of poetry in life
CO3: Develop aesthetic sense
CO4: Develop holistic perspective to their personality

Syllabus
Practical 1: **Art of poetry – orientation**
Practical 2: **Forms of poetry – orientation**
Practical 3: **Forms of poetry – recitation**
Practical 4: **Application of poetry – orientation**
Practical 5: **Application of poetry – practical session**
Practical 6: **Poetry and aesthetics**
Practical 7: **Writing poetry – orientation**
Practical 8: **Writing poetry – writing sessions**

Reading material

I. The Art of Poetry
1. Fry, S. (2005). The ode less travelled: Unlocking the poetic mind. HarperCollins.
2. Addonizio, K., & Laux, D. (1997). The poet's companion: A guide to the pleasures of writing poetry. W.W. Norton & Company.
3. Lucy, J. (Ed.). (2001). The art of poetry. Penguin Books.

II. Understanding and Interpretation of Poetry
1. Hirsch, E. (1999). How to read a poem: And fall in love with poetry. Harcourt Brace & Company.
2. Pinsky, R. (1998). The sounds of poetry: A brief history. Farrar, Straus and Giroux.
3. Meyer, M. (2005). Poetry: An introduction. Bedford/St. Martin's.

III. Writing Poetry
1. Hugo, R. (1979). The triggering town: Lectures and essays on poetry and writing. W.W. Norton & Company.
2. Bradbury, R. (1990). Zen in the art of writing: Releasing the creative genius within you. Bantam Books.
3. Behn, R., & Twichell, C. (Eds.). (1992). The practice of poetry: Writing exercises from poets who teach. HarperCollins

Course Code	25HS02PR0102-14			
Category	Co-Curricular Activity			
Course Title	Creative and Content Writing			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course objective:

The objective of the course is to equip students with comprehensive skills in creative and content writing through experiential learning and real-world applications.

Course outcomes:

On completion of the course, student will be able to achieve the following:

CO1: Understand and apply fundamental concepts and techniques of creative writing.

CO2: Apply storytelling techniques to create engaging narratives.

CO3: Develop and implement effective SEO and digital content strategies

CO4: Create and refine content using various tools and applying diverse writing styles and formats.

CO5: Utilize digital tools to craft multimedia narratives and create a professional portfolio.

Syllabus

Creative Writing

Practical 1: **Introduction to Creative and Content Writing**

Practical 2: **Character and Story Development**

Practical 3: **Crafting Compelling Narratives**

Content Writing

Practical 4: **SEO and Digital Content Strategies**

Practical 5**: Writing for Media**

Practical 6: **Tools**

Content Creation

Practical 7: **Digital Storytelling**

Practical 8: **Creative Portfolio Launch**

Course Code	25HS02PR0102-15			
Category	Co-Curricular Activity			
Course Title	Science of life through Bhagwad Gita			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Objective
The objective of the course is to seek directions from the Bhagwad Gita to garner life skills for a successful and happy life

Course Outcome
CO1: To understand the methodology to correctly interpret and analysis the scripture
CO2: To understand the application of various teaching of the Bhagwad Gita
CO3: Use meditation and breathing techniques for healthy mind and body.

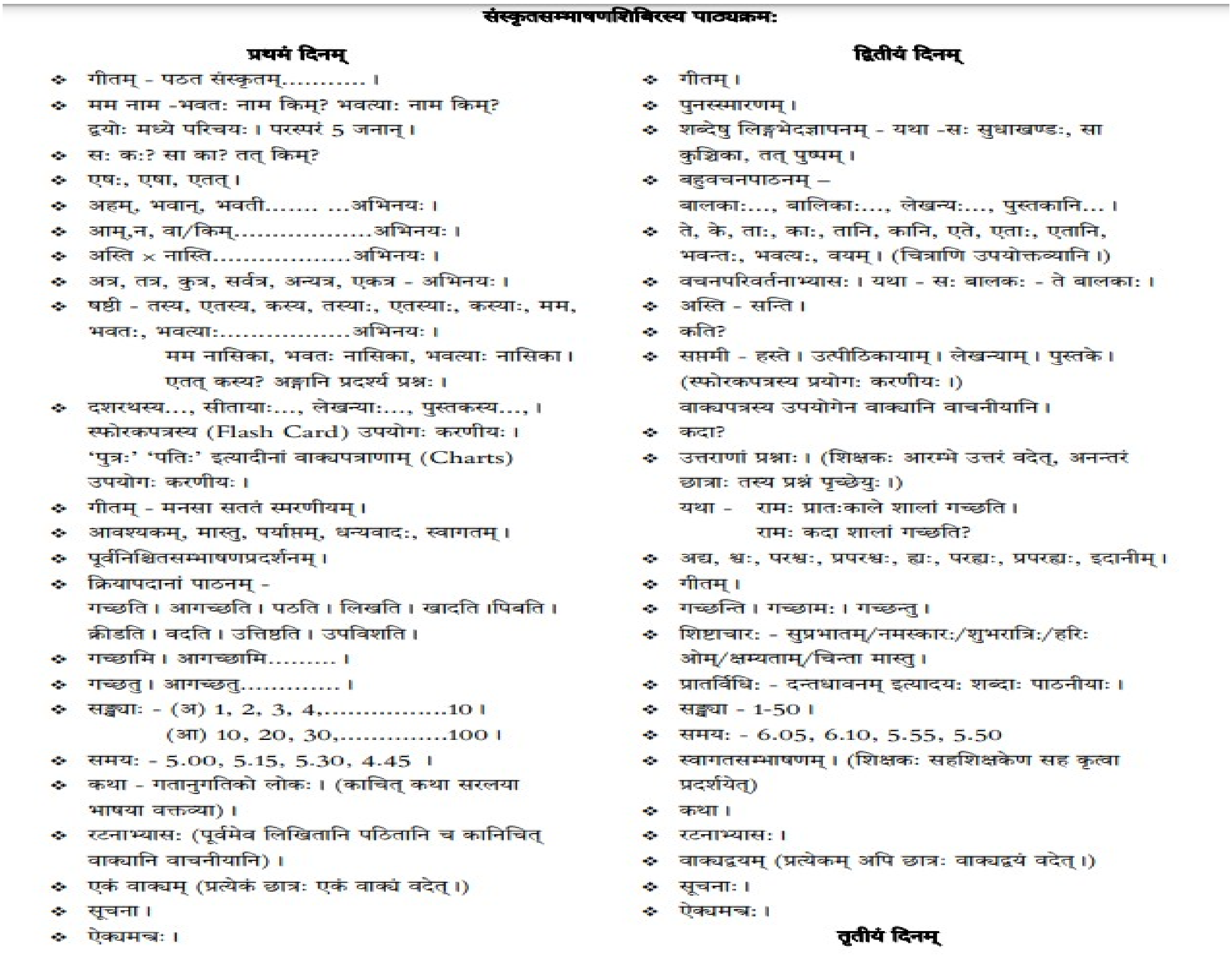
Syllabus
Practical 1: **Introduction to Bhagwad Gita - methodology**
Practical 2: **Real life application of chapter 1-3**
Practical 3: **Real life application of chapter 4-6**
Practical 4: **Real life application of chapter 7-9**
Practical 5: **Real life application of chapter 10-12**
Practical 6: **Real life application of chapter 13-15**
Practical 7: **Real life application of chapter 16-18**
Practical 8: **Meditation and breathing techniques**

Course Code	25HS02PR0102-16			
Category	Co-Curricular Activity			
Course Title	Sanskrit Sambhashan- Spoken Sanskrit			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course objectives:
The objective of the course is to enhance the communication skills of the students in Sanskrit

Course outcome
At the end of the course, the students will be able to achieve the following:
CO1: Enhanced writing skills in Sanskrit
CO2: Enhanced speaking skills in Sanskrit
CO3: Enhanced listening skills in Sanskrit
CO4: Enhanced writing skills in Sanskrit

Syllabus:



संस्कृतसम्भाषणशिबिरस्य पाठ्यक्रमः

प्रथमं दिनम्

- गीतम् - पठत संस्कृतम्........... ।
- मम नाम -भवतः नाम किम्? भवत्याः नाम किम्?
  द्वयोः मध्ये परिचयः । परस्परं 5 जनान् ।
- सः कः? सा का? तत् किम्?
- एषः, एषा, एतत् ।
- अहम्, भवान्, भवती....... ...अभिनयः ।
- आम्,न, वा/किम्...................अभिनयः ।
- अस्ति × नास्ति..................अभिनयः ।
- अत्र, तत्र, कुत्र, सर्वत्र, अन्यत्र, एकत्र - अभिनयः ।
- षष्ठी - तस्य, एतस्य, कस्य, तस्याः, एतस्याः, कस्याः, मम, भवतः, भवत्याः.................अभिनयः ।
  मम नासिका, भवतः नासिका, भवत्याः नासिका ।
  एतत् कस्य? अङ्गानि प्रदर्श्य प्रश्नः ।
- दशरथस्य..., सीतायाः...., लेखन्याः...., पुस्तकस्य..., ।
  स्फोरकपत्रस्य (Flash Card) उपयोगः करणीयः ।
  'पुत्रः' 'पतिः' इत्यादीनां वाक्यपत्राणाम् (Charts) उपयोगः करणीयः ।
- गीतम् - मनसा सततं स्मरणीयम् ।
- आवश्यकम्, मास्तु, पर्याप्तम्, धन्यवादः, स्वागतम् ।
- पूर्वनिश्चितसम्भाषणप्रदर्शनम् ।
- क्रियापदानां पाठनम् -
  गच्छति । आगच्छति । पठति । लिखति । खादति ।पिबति । क्रीडति । वदति । उत्तिष्ठति । उपविशति ।
- गच्छामि । आगच्छामि......... ।
- गच्छतु । आगच्छतु............. ।
- सङ्ख्याः - (अ) 1, 2, 3, 4,.................10 ।
  (आ) 10, 20, 30,..............100 ।
- समयः - 5.00, 5.15, 5.30, 4.45 ।
- कथा - गतानुगतिको लोकः । (काचित् कथा सरलया भाषया वक्तव्या) ।
- रटनाभ्यासः (पूर्वमेव लिखितानि पठितानि च कानिचित् वाक्यानि वाचनीयानि) ।
- एकं वाक्यम् (प्रत्येकं छात्रः एकं वाक्यं वदेत् ।)
- सूचना ।
- ऐक्यमन्त्रः ।

द्वितीयं दिनम्

- गीतम् ।
- पुनस्स्मारणम् ।
- शब्देषु लिङ्गभेदज्ञापनम् - यथा -सः सुधाखण्डः, सा कुञ्चिका, तत् पुष्पम् ।
- बहुवचनपाठनम् –
  बालकाः...., बालिकाः...., लेखन्यः...., पुस्तकानि... ।
- ते, के, ताः, काः, तानि, कानि, एते, एताः, एतानि, भवन्तः, भवत्यः, वयम् । (चित्राणि उपयोक्तव्यानि ।)
- वचनपरिवर्तनाभ्यासः । यथा - सः बालकः - ते बालकाः ।
- अस्ति - सन्ति ।
- कति?
- सप्तमी - हस्ते । उत्पीठिकायाम् । लेखन्याम् । पुस्तके ।
  (स्फोरकपत्रस्य प्रयोगः करणीयः ।)
  वाक्यपत्रस्य उपयोगेन वाक्यानि वाचनीयानि ।
- कदा?
- उत्तराणां प्रश्नाः । (शिक्षकः आरम्भे उत्तरं वदेत्, अनन्तरं छात्राः तस्य प्रश्नं पृच्छेयुः ।)
  यथा - रामः प्रातःकाले शालां गच्छति ।
  रामः कदा शालां गच्छति?
- अद्य, श्वः, परश्वः, प्रपरश्वः, ह्यः, परह्यः, प्रपरह्यः, इदानीम् ।
- गीतम् ।
- गच्छन्ति । गच्छामः । गच्छन्तु ।
- शिष्टाचारः - सुप्रभातम्/नमस्कारः/शुभरात्रिः/हरिः ओम्/क्षम्यताम्/चिन्ता मास्तु ।
- प्रातर्विधिः - दन्तधावनम् इत्यादयः शब्दाः पाठनीयाः ।
- सङ्ख्या - 1-50 ।
- समयः - 6.05, 6.10, 5.55, 5.50
- स्वागतसम्भाषणम् । (शिक्षकः सहशिक्षकेण सह कृत्वा प्रदर्शयेत्)
- कथा ।
- रटनाभ्यासः ।
- वाक्यद्वयम् (प्रत्येकम् अपि छात्रः वाक्यद्वयं वदेत् ।)
- सूचनाः ।
- ऐक्यमन्त्रः ।

तृतीयं दिनम्

- गीतम्।
- पुनस्स्मारणम्।
- क्रियापदानां बहुवचनरूपाणि।

शृणोमि – शृणुमः। शृणोति - शृण्वन्ति।

जानामि – जानीमः। जानाति – जानन्ति।

- सम्बोधनम् - भोः !, श्रीमन् !, मान्ये !, भगिनि!, मित्र !,

- भोजनसम्बन्धिशब्दाः यथा - सूपः, शाकम्, इत्यदयः।
- सङ्ख्या।
- समयः।

- बहु x किञ्चित्,
- दीर्घः x ह्रस्वः।
- उन्नतः x वामनः।

- गीतम्।
- आरोग्यसम्बन्धिशब्दाः – वैद्यरोगिसम्भाषणम्।
- प्रश्नोत्तरस्पर्धा।
- ऋषीणां नामानि।
- कथा - शिक्षकः एकां कथां वदति। अनन्तरं छात्रेषु एकैकः तस्याः कथायाः एकैकं वाक्यम् उक्त्वा कथां सम्पूर्णां करोति।
- सङ्ख्या - दीर्घसङ्ख्यापाठनम्।
- प्रश्नोत्तरम्।
- क्रीडा - (गणद्वये नामस्मरणक्रीडा)
- कथा।
- पुस्तकानां परिचयः।
- सप्त वाक्यानि।
- सूचना।
- ऐक्यमन्त्रः।

अष्टमं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- वारम्।
- अतः - यतः परिवर्तनाभ्यासः।
- यद्यपि – तथापि।
- यत्र – तत्र।
- कति - कियत् - एतयोः भेदज्ञापनम्।
- यावत् – तावत्।
- यत् – तत्।
- यः - सः।
- या – सा।
- गीतम्।
- अस्माकम्।
- चर्चा।
- सङ्ख्या - 'शतायुः - गतायुः' क्रीडा।
- विनोदकणिकाकथनम्।
- कथा।
- अष्ट वाक्यानि।
- समाजनिधिविषये सूचना।
- ऐक्यमन्त्रः।

नवमं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- चित्।
- ....द्वयम्।
- सङ्ख्यासु लिङ्गभेदः।

एकः - एका - एकम्

द्वयम् - द्वयम् - द्वयम्

त्रयः - तिस्रः - त्रीणि

चत्वारः - चतस्रः - चत्वारि

- शिश्रकः - अहं वैद्यः - मम नाम सुरेशः
  (छात्राः तमुद्दिश्य प्रश्नान् पृच्छेयुः।)
- .... अर्थम् (समाजार्थम्, संस्कृतकार्यार्थम्...)।
- गीतम्।
- तव्यत् – अनीयर्।
- अनन्त्यकथारचना।
- सङ्ख्यान्वेषणम् (क्रीडा)।
- छात्रैः सह प्रश्नोत्तरम्।
- समाजनिधिविषये पुनःस्मारणम्।
- ऐक्यमन्त्रः।

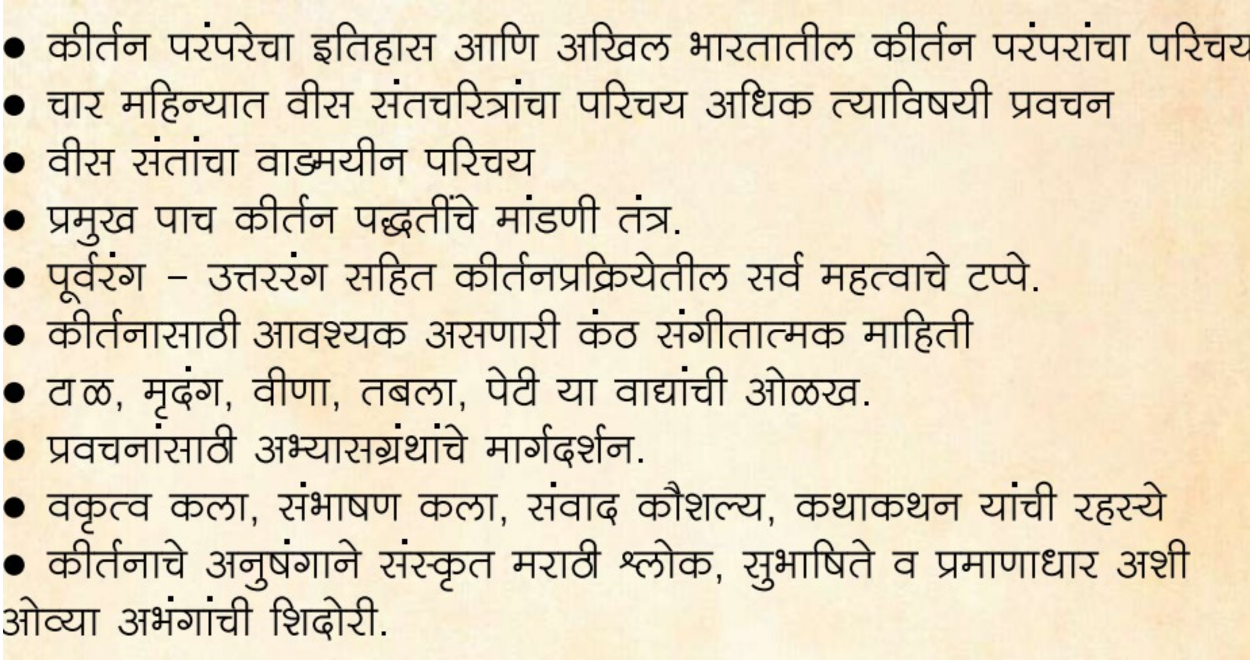
दशमं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- पत्रलेखनम्।
- दूरवाणीसम्भाषणम्।
- मार्गनिर्देशः – कुत्र गन्तव्यम् इत्यादि।
- तव्यत् अभ्यासार्थम् – अद्य किं किं करणीयम् ?
- सान्दर्भिकभाषणम् –
  1. प्रवासात् प्रतिनिवर्तनस्य।
  2. आपणिकस्य इत्यादि।
- क्रीडा – सङ्ख्यायोजनम् (गणद्वये)।
- शुभाशयाः।
- असत्यकथनम् / कल्पनाकथनम्।
- समारोपः (सर्वैः शिक्षार्थिभिः भारतमातुः पूजां कृत्वा निधिसमर्पणं करणीयम्।)
- पत्राचारप्रगतशिक्षणादिविषये सूचना।
- ऐक्यमन्त्रः।

Course Code	25HS02PR0102-17			
Category	Co-Curricular Activity			
Course Title	Kirtan Kala			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course objectives:
The objective of the course is to provide the students with a spiritual experience as well as its benefits to them in the form of better abilities to concentrate and develop the ability to create a peaceful mind.

Course outcome
At the end of the course, the students will be able to achieve the following:
CO1: Learn from the inspiring spiritual journey of the saints and the history of Kirtan tradition
CO2: Learn about the musical instruments used in the art of Kirtan
CO3: Develop communication skills



- कीर्तन परंपरेचा इतिहास आणि अखिल भारतातील कीर्तन परंपरांचा परिचय
- चार महिन्यात वीस संतचरित्रांचा परिचय अधिक त्याविषयी प्रवचन
- वीस संतांचा वाङ्मयीन परिचय
- प्रमुख पाच कीर्तन पद्धतींचे मांडणी तंत्र.
- पूर्वरंग – उत्तररंग सहित कीर्तनप्रक्रियेतील सर्व महत्वाचे टप्पे.
- कीर्तनासाठी आवश्यक असणारी कंठ संगीतात्मक माहिती
- टाळ, मृदंग, वीणा, तबला, पेटी या वाद्यांची ओळख.
- प्रवचनांसाठी अभ्यासग्रंथांचे मार्गदर्शन.
- वकृत्व कला, संभाषण कला, संवाद कौशल्य, कथाकथन यांची रहस्ये
- कीर्तनाचे अनुषंगाने संस्कृत मराठी श्लोक, सुभाषिते व प्रमाणाधार अशी ओव्या अभंगांची शिदोरी.

Course Code	25HS02PR0102-18			
Category	Co-Curricular Activity			
Course Title	Introduction to German Language and culture			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course objective:

To help build a foundation and interest in German language so that the students can pursue the proficiency levels of the language in higher semesters.

Course outcomes:

On successful completion of the course the students will be able to achieve the following:
CO1. Demonstrate basic knowledge about Germany, the culture and similarities/differences between India and Germany
CO2. Learn to use simple language structures in everyday communication.
CO3. Develop ability to write in basic German about themselves and others.
CO4. Develop ability to read and understand beginner level texts in German.

Syllabus
List of Practicals

Practical-1: Orientation about Germany, the language, and culture
Practical-2: Vocabulary building for everyday conversations
Practical -3: Numbers, days and time
Practical-4: Introducing Oneself & Others
Practical-5: Reading Skills: Reading simple text in German language
Practical-6: Basic Verbs & Sentence Construction
Practical-7: Food & Dining, Giving Directions & Transportation
Practical-8: Writing Skills: Write basic German and practice

Recommended reading

1. German Made Easy by Diego A. Agundez
2. Teach Yourself Complete German: Learn to Read, write, Speak and Understand A new Language by Paul Coggle, Heiner Schenke
3. Netzwerk A1 by Helen Smitz, Stefanie Dengler and Paul Rusch
4. Deutsche Welle (DW) – www.dw.com/learngerman
5. BBC Languages – German – www.bbc.co.uk/languages/german
6. Goethe-Institut – www.goethe.de

Course Code	25HS04PR0102-1			
Category	Co-Curricular Activity			
Course Title	Adventure Sports			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Objective:

This course introduces adventure sports, emphasizing experiential learning through participation in various activities. The course will cover the fundamentals, safety procedures, and physical and mental benefits of adventure sports. Students will engage in outdoor activities such as wall climbing, rappelling, and more, fostering a connection with nature and understanding the principles of risk management.

Course Outcome: By the end of this course, students will:

- Understand the principles and benefits of adventure sports.
- Develop basic skills in selected adventure sports.
- Learn and apply safety measures and risk management techniques.
- Foster teamwork, leadership, and problem-solving skills.
-

Syllabus:

- Tent pitching, knot practice session and Tent allotment
- Activities like Jumarring and Climbing
- Individual challenge like Burma bridge, ladder bridge, multi vine
- Group Task like improvise raft making and Kayaking
- Activities like Archery rifle shooting, cycle ride

Pattern of Classes: 30 Hrs. Camp (2 Days and 1 Night)

Assessment Pattern:

Assessment Type	Weightage in Marks	Total Marks
Practical 25HS04PR0102-1	Internal Marks – 25 Marks External Marks – 25 Marks	50
		Total – 50 Marks

Course Code	25HS04PR0102-2			
Category	Co-Curricular Activity			
Course Title	Introduction to Defense Forces & Obstacle Training			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Objective:

- Understand the Structure and Function of Defense Forces
- Familiarize with Defense Force Training and Discipline
- Learn Basic Obstacle Course Techniques
- Apply Problem-Solving and Teamwork in Obstacle Training
- Explore the Role of Obstacle Training in Defense Preparedness
-

Course Outcome:

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

- Describe the Structure and Functions of Defense Forces
- Demonstrate Knowledge of Defense Training Protocols
- Navigate Basic Obstacle Courses & Connect Obstacle Training to Defense Preparedness
- Collaborate and Problem-Solve in Team-Based Exercises

Syllabus:

- Knot and Hitch practice session
- Activities like Rappelling & Wall Climbing
- Burma bridge & ladder bridge
- First Aid
- Rifle Shooting
- Horse riding
- Group Task and Team building activities

Pattern of Classes: 30 Hrs Camp (2 Days and 1 Night Camp)

Assessment Pattern:

Assessment Type	Weightage in Marks	Total Marks
Practical 25HS04PR0102-2	Internal Marks – 25 Marks External Marks – 25 Marks	50
	Total – 50 Marks	

Course Code	25HS04PR0102-3			
Category	Co-Curricular Activity			
Course Title	First Aid & Disaster Management			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Objective:

- Understand Disaster Types and Characteristics
- Learn Risk Assessment and Management
- Master Emergency Preparedness and Response
- Explore Recovery and Reconstruction
- Develop Skills in Communication and Coordination
- Understand Legal and Ethical Considerations

Course Outcome:

Upon successful completion of the disaster management course, students should be able to:

- Identify and Categorize Disasters
- Conduct Risk Assessments
- Develop Emergency Plans
- Implement Response Strategies

Syllabus:

- Basic First Aid
- Transportation of Casualty
- Injury Prevention & Cure
- Various Types of Knots & Hitches
- Various team building activities
- Fire emergencies & use of extinguishers (Optional)
- Snake Bite & Environmental emergencies.

Assessment Pattern:

Assessment Type	Weightage in Marks	Total Marks
Practical 25HS04PR0102-3	Internal Marks – 25 Marks External Marks – 25 Marks	50
		Total – 50 Marks

Course Code	25HS04PR0102-4			
Category	Co-Curricular Activity			
Course Title	Basic Nutritional Course			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Objective:

In the "Basics of Nutrition" course, students will develop a comprehensive understanding of essential nutrients and their roles in supporting overall health. They will learn to apply dietary guidelines effectively, tailoring recommendations to various age groups and health conditions. Additionally, students will cultivate the skills needed to assess and improve their own and others' eating habits for better health outcomes.

Course Outcome:

By the end of the course, students will be able to

- Accurately describe the functions of key nutrients and their impact on health,
- create balanced meal plans based on established dietary guidelines,
- critically evaluate nutrition information to distinguish between credible and misleading sources.

Syllabus:

Unit I

- Introduction to Nutrition – Define Balanced Diet, Nutrition, Optimum Nutrition, Nutrients, Concept of Health, Recommended Dietary Allowances (RDA)
- Carbohydrates (sources, functions and digestion)
- Proteins (sources, functions and digestion)
- Fats (sources, functions and digestion)
- Micronutrients (vitamins and minerals-sources, functions and digestion)

Practical I

- Display of all the foods with the help of students and while demonstrating teacher will again explain role and importance of nutrition in daily life. Deficiency will lead to chronic diseases and its prevention is very necessary for the quality of life.

Unit II

- What is Body Mass Index?
- What is Basal Metabolic Rate?
- What is Ideal Body Weight? (Male/Female)
- How to read labels on Food Packets?
- How to choose smart food and Concept of Rainbow diet, My Food Pyramid or My plate given by ICMR- NIN.

Practical II

- Calculation of Body Mass Index, Basal Metabolic Rate, Ideal Body Weight (Male/Female) with the use of self-body measurements.
- Demonstration of Rainbow diet, My Food Pyramid or My plate in a class.

Assessment Pattern:

Assessment Type	Weightage in Marks	Total Marks
Practical 25HS04PR0102-4	Internal Marks – 25 Marks External Marks – 25 Marks	50
		Total – 50 Marks

Course Code	25HS04PR0102-5			
Category	Co-Curricular Activity			
Course Title	Stress Management Through Yoga & Meditation			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Objective: Mental health is one of the most important facets of human life. Academic learning has emerged as a major source of stress among young students worldwide. Promoting mental well-being among students in India is a crucial step toward achieving Sustainable Development Goal 3 (Good Health and Well-being). Stress management involves using various techniques and strategies to control stress levels, improve how you react to stressful situations. Yoga combines physical movement with deep breathing and meditation, providing a holistic approach to stress relief.

Course Outcome:

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

● Understand the basics of stress management.

● Analyze stress triggers and to manage them.

● Evaluate the responses to stressful situations.

● Apply the techniques of Yog & Meditation for stress management in day-to-day life.

Syllabus:

Unit-1

Introduction to Stress: The Meaning of Stress, types of stress: distress, eustress

Stress Management Techniques I:

Treatment 1- (Asanas): Tadasana, Trikonasana, Vrikshasana, Garudasana,, Ardha-Padamasana, Padamasana, Vajarasana, Ushtrasana, Gomukhasna,, Paschimottanasan, ,Ardha Halasana, ,Setu-Bandhanasa,Naukasana, Bhujangasana, and Dhanurasana; along with relaxing asanas

Unit-2

Spiritual approach to stress management.

Stress Management Techniques II

Treatment 2 – (Prananyam) Deep breathing, Yoga, Mindfulness meditation

Rechak, Purak, Kumbhak, Nadi Suddhi and Bhramari Pranayama.

Measuring Academic stress- It can be measure using questionnaire: Academic stress Scale (Sun .et al 2011**).**

Assessment Pattern:

Assessment Type	**Weightage in Marks**	**Total Marks**
Practical 25HS04PR0102-5	Continuous Assessment – 25 Marks Internal Test Evaluation – 25 Marks	50
		Total – 50 Marks

SYLLABUS OF SEMESTER II

Course Code	25EE01TP0201			
Category	Engineering Science Course			
Course Title	Microcontroller and Interfacing			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	II

Course Outcomes

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

- Understand the architecture of microprocessor and microcontroller.
- Develop, understand and analyze programming of a microprocessor and microcontroller.
- Acquire the knowledge, techniques and skill to interface external peripheral devices with microprocessor or microcontroller.
- Design microcontroller-based system to solve the real-world problem

Syllabus

Module I: **(06 Hrs)**
Introduction to RISC and CISC processors, Harvard and Von Neumann architecture, Introduction to Intel's 8086 architecture, pin diagram, bus concepts, addressing modes, segmentation, pipelining.

Module II:(06 Hrs)
Instruction set, stack and subroutines- simple and nested, stack manipulation, Memory mapping, interrupts-concept and structure, interrupt service routines, simple programs.

Module III: (06 Hrs)
Introduction to x51 Family Microcontrollers, their Architecture,pindiagram,addressing mode SFRs.

Module IV:(06 Hrs)
Instruction of 8051 , stack and subroutines- Memory concept, interrupts-concept and structure, interrupt service routines, Sleep mode, idle mode, Run Mode. simple programming.

Module V:(06 Hrs)
Interfacing of Switches & Relays, Stepper motor, LED, SSD, LCD, Analog-to-Digital Converter (ADC), DC motor with x51controller.

Module 6: **(05 Hrs)**
Commutation Protocols used with microcontroller: Parallel communication, Serial communication, Serial Peripheral Interface (SPI), I2C Communication, Introduction to USB.

Textbook:

1.Advanced Microprocessors and Peripherals; A. K. Ray & K. M. Bhurchandi; McGraw Hill, 3rdEdition

2.The 8051 Microcontroller and Embedded Systems Using Assembly and C; Muhammad Ali Mazidi, Pearson, 2nd Edition

Reference Books:

1.Microcomputer systems: the 8086/8088 family: Architecture, Programming, and Design; Yu-chengnd Liu, Glenn A. Gibson; Prentice-Hall, 2nd Edition.

2.The 8051 Microcontroller: Architecture, Programming, and Applications Kenneth Ayala, Pearson, 3rd Edition.

LIST OF EXPERIMENTS

PART-A:

Conduct the following experiments by writing Assembly Language Program (ALP) using 8086 and 8051 Registers using an evaluation board/simulator and the required software tool.

1.Write an ALP to multiply two 16-bit binary numbers.
2.Write an ALP to find the sum of first 10 integer numbers
3. Write an ALP to find factorial of a number.
4. Write an ALP to add an array of 16-bit numbers and store the 32-bit result in internal RAM
5. Write an ALP to add two 64-bit numbers.
6. Write an ALP to find the square of a number (1 to 10) using look-up table.
7. Write an ALP to find the largest/smallest number in an array of 32 numbers.
8. Write an ALP to arrange a series of 32-bit numbers in ascending/descending order.
9. Write an ALP to count the number of ones and zeros in two consecutive memory locations.
10. Write an ALP to Scan a series of 32-bit numbers to find how many are negative.

PART-B:

Conduct the following experiments on 8086/8051 evaluation board using evaluation version of Embedded 'C' & Keil compiler.

1. Display “Hello World” message using Internal UART.
2. Interface and Control a DC Motor.
3. Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction.
4. Determine Digital output for a given Analog input using Internal ADC of ARM controller.
5. Interface a DAC and generate Triangular and Square waveforms.
6. Interface a 4x4 keyboard and display the key code on an LCD.
7. Using the Internal PWM module of ARM controller generate PWM and vary its duty cycle.
8. Demonstrate the use of an external interrupt to toggle an LED On/Off.
9. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.
10. Interface a simple Switch and display its status through Relay, Buzzer and LED.

Course Code	25HS03TH0213			
Category	Basic Science Course			
Course Title	Calculus & Linear Algebra			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	II

Course Objective

The objective of this course is to familiarize the prospective engineers with techniques in Calculus. It aims to equip the students with standard concepts and tools at an intermediate to advanced level that will serve them well towards tackling more advanced level of mathematics and applications that they would find useful in their disciplines.

Course Outcomes

On successful completion of the course, student shall be able to

1. Apply the concepts of continuity and differentiability to find Taylor's and Maclaurin series.
2. Understand the methods of partial derivatives and apply these concepts to determine extreme values of the functions of two variables.
3. Demonstrate the basic knowledge of vector differentiation and line integral.
4. Interpret the solutions of system of linear equations and use the concepts of Eigen values, Eigen vectors to find diagonalization of matrices, reduction of quadratic form to canonical form.
5. Internalize convergence of sequences and apply it to determine whether infinite series convergent or divergent with appropriate tests.

Syllabus

Module 1: (8 Lectures)

Differential Calculus: Functions of univariate, Mean value theorems: Rolle's theorem, Lagrange's theorem, Cauchy's theorem, Taylor's theorem, Taylor's and Maclaurin series.

Module 2: (10 Lectures)

Partial Differentiation: Partial derivatives, Euler's Theorem, chain rule, application of partial differentiation: total derivative, Jacobians, Maxima, Minima for the functions of two variables., Extrema of function of multivariable,

Module 3: (8 Lectures)

Vector Calculus: Scalar and vector fields, gradient of scalar point function, directional derivatives, divergence and curl of vector point function, application of vector calculus: Line integral, Gradient Descent method.

Module 4: (8 Lectures)

Rank-nullity theorem; Consistency of system of linear equations and its solution, Orthogonal matrices, Eigen values and eigenvectors, Diagonalization of matrices, Orthogonal transformation and quadratic to canonical forms, Introduction to n-dimensional vector spaces, Singular value decomposition and its applications.

Module 5: (6 Lectures)

Infinite series: Sequences, Infinite series of real and complex numbers, Cauchy criterion, tests of convergence, absolute and conditional convergence, uniform convergence, power series, radius of convergence.

Textbooks/References

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. Veerarajan T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
4. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
5. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.
6. P. N. Wartikar and J. N. Wartikar, A text book of Applied Mathematics Volume I & II, Pune Vidhyarthi Griha Prakashan, Pune-411030 (India).

Course Code	25EE01TP0202			
Category	Program Core Course			
Course Title	Programming for Problem Solving			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	II

Course Outcomes

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

1. Develop the fundamentals of C programming and choose the loops and decision-making statements to solve and execute the given problem.
2. Formulate simple algorithms for arithmetic and logical problems, translate the algorithms to programs, test and execute the programs, and correct syntax and logical errors.
3. Use arrays, pointers, structures, and I/O operations for the formulation of algorithms and programs.
4. Apply programming concepts to solve matrix addition, multiplication problems, and searching & sorting problems.
5. Implement iterations and recursions, decompose a problem into functions, and synthesize a complete program using divide and conquer approach.

Syllabus

Module I:
Introduction to Programming: Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.).
Idea of Algorithm: Steps to solve logical and numerical problems.
Representation of Algorithm: Flowchart/Pseudocode with examples.
Arithmetic expressions and precedence.

Module II:
C Programming Language: Introduction to C language: Keywords, Constant, Variable, Data types, Operators, Types of Statements, Pre-processor Directives,
Decision Control Statement - if, if-else, nested if-else statement, switch case, Loops, and writing and evaluation of conditionals and consequent branching.

Module III:
Arrays and Basic Algorithms: Arrays: 1-D, 2-D, Character arrays and Strings.
Searching, Basic Sorting Algorithms, Finding roots of equations, notion of order of complexity through example programs (no formal definition required).

Module IV:

Functions and Recursion: User-defined and Library Functions, Parameter passing in functions, call by value, passing arrays to functions: idea of call by reference.
Recursion: As a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series.

Module V:

Pointers and Structures: Structures, Defining structures, Array of Structures, Introduction to pointers, Defining pointers, Pointer arithmetic, Pointer operators, Use of Pointers in self-referential structures.

Module VI:

File Handling: Streams in C, Types of Files, File Input/Output Operations: Modes of file opening, Reading and writing the file, Closing the files using fflush().

Text Books

1. *Programming in ANSI C* – E. Balagurusamy, McGraw Hill
2. *Mastering C* – K. R. Venugopal and S. R. Prasad, Tata McGraw Hill

Reference Books

1. *Programming with C* – Byron Gottfried, Schaum's Outline Series
2. *Let Us C* – Yashwant Kanetkar, BPB Publication

. **LIST OF EXPERIMENTS**

1. Using basic data types of C, implement arithmetic expressions.
2. Implement programs using Decision Control Structures.
3. Demonstrate use of Loop Control Structures.
4. Implement programs using Multi-way Decision Control Structures (Switch Case).
5. Apply Functions and Recursion to simplify programs.
6. Initialize arrays and apply them to solve problems of 1D and 2D arrays.
7. Demonstrate use of Structures and Pointers.
8. Apply file handling concepts in C.

Note: 2 to 3 practice programs will be taken on each of the experiments mentioned above.

Course Code	25EE01TH0203			
Category	Engineering Science Course			
Course Title	AI and ML Essentials			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	II

Course Description:

The AI and ML Essentials course provides a comprehensive introduction to the fundamental concepts and applications of Artificial Intelligence (AI) and Machine Learning (ML). This 45-hour course covers key topics such as AI principles, machine learning techniques (supervised, unsupervised, reinforcement learning), natural language processing, data handling, popular AI/ML tools and platforms, and ethical considerations. The course also explores future trends in AI and ML, including deep learning, AI in robotics, healthcare, and automation.

Course Objectives:

- Understand the core concepts and principles of Artificial Intelligence (AI).
- Learn the basics of Machine Learning (ML) and its different types (supervised, unsupervised, reinforcement learning).
- Explore the key components of AI and ML, including data, algorithms, and computing power.
- Gain insights into Natural Language Processing (NLP) and its applications.
- Become familiar with popular AI/ML libraries and frameworks, particularly Python libraries like NumPy, Pandas, Scikit-learn, TensorFlow, and Keras.
- Understand the ethical considerations and challenges in AI, including bias, transparency, and accountability.
- Explore future trends in AI, such as deep learning, AI in robotics, healthcare, and automation.

Course Outline:

Unit 1: Overview of Artificial Intelligence (AI) - 8 hours

Definition of AI and Key Concepts

- **Understanding Artificial Intelligence**
  - Definition: AI refers to the simulation of human intelligence in machines programmed to think, learn, and make decisions.
  - Core Principles: Reasoning, problem-solving, perception, learning, and language understanding.
  - Difference between AI, Machine Learning (ML), and Deep Learning (DL).
- **Types of AI**
  - **Narrow AI (Weak AI):** Designed for specific tasks (e.g., facial recognition, chatbots).
  - **General AI (Strong AI):** Hypothetical AI with human-like cognitive abilities.
  - **Super AI:** AI surpassing human intelligence and capabilities.
- **Key Components of AI**
  - **Data:** The foundation of AI systems; types of data (structured, unstructured, semi-structured).
  - **Algorithms:** Rules and statistical models used to process data (e.g., decision trees, neural networks).
  - **Computing Power:** Role of GPUs, TPUs, and cloud computing in enabling AI.

Historical Perspective and Evolution of AI

- **Key Milestones in AI Development**
  - 1950: Alan Turing's Turing Test.
  - 1956: The Dartmouth Conference (birth of AI as a field).
  - 1997: IBM Deep Blue defeats chess champion Garry Kasparov.
  - 2011: IBM Watson wins *Jeopardy!*.

    - 2016: AlphaGo defeats world champion Go player.
- **Early AI Systems vs Modern AI**
    - Early AI: Rule-based systems, limited data, and computing power.
    - Modern AI: Data-driven approaches, deep learning, and scalable infrastructure.

Applications of AI in Everyday Life

- **AI in Personal Assistants**
    - How Siri, Alexa, and Google Assistant use Natural Language Processing (NLP) and speech recognition.
- **AI in Autonomous Vehicles**
    - Role of computer vision, sensor fusion, and reinforcement learning.
- **AI in Healthcare**
    - Applications in diagnostics, drug discovery, and personalized medicine.
- **AI in E-commerce**
    - Recommendation systems, fraud detection, and customer support chatbots.

Machine Learning Basics

- Supervised Learning: Training models with labeled data (e.g., spam detection).
- Unsupervised Learning: Finding patterns in unlabeled data (e.g., customer segmentation).
- Reinforcement Learning: Learning through trial and error (e.g., game-playing AI).

Natural Language Processing (NLP)

- Definition: A subfield of AI focused on interaction between computers and humans using natural language.
- Key Concepts:
    - Tokenization: Breaking text into words or phrases.
    - Sentiment Analysis: Determining the emotional tone of text.
    - Named Entity Recognition (NER): Identifying names, dates, and other entities.
- Applications: Chatbots, voice assistants, and language translation.

AI Ethics and Bias

- Importance of unbiased data and algorithms.
- Case studies of AI bias in facial recognition and hiring systems.

Future Trends in AI

- Explainable AI (XAI): Making AI decisions transparent and interpretable.
- AI in edge computing: Bringing AI capabilities to local devices.
- Quantum computing and its potential impact on AI.

Challenges and Opportunities in AI

- **Challenges**
    - Ethical concerns: Bias, privacy, and job displacement.
    - Technical limitations: Data quality, interpretability, and scalability.
- **Opportunities**
    - Advancements in healthcare, education, and sustainability.
    - Potential for solving complex global problems.

Unit 2: Introduction to Machine Learning (ML) - 9 hours

What is Machine Learning?

- **Basic Concepts of Machine Learning**
    - Definition of Machine Learning (ML): Teaching machines to learn from data without explicit programming.
    - Key components: Data, algorithms, models, and predictions.
    - Difference between traditional programming and ML.

    - Applications of ML in real-world scenarios.
- **Types of Machine Learning: Supervised, Unsupervised, and Reinforcement Learning**
    - **Supervised Learning**:
        - Definition: Learning from labeled data (input-output pairs).
        - Examples: Classification (e.g., spam detection) and regression (e.g., house price prediction).
    - **Unsupervised Learning**:
        - Definition: Learning from unlabeled data to find patterns or groupings.
        - Examples: Clustering (e.g., customer segmentation) and dimensionality reduction (e.g., PCA).
    - **Reinforcement Learning**:
        - Definition: Learning through trial and error using rewards and penalties.
        - Examples: Game playing (e.g., AlphaGo), robotics, and autonomous driving.

Relation between AI and ML

- **The Role of ML in AI Systems**
    - AI as the broader concept: Machines performing tasks that require human intelligence.
    - ML as a subset of AI: Enabling systems to learn and improve from experience.
    - Deep learning as a subset of ML: Neural networks for complex tasks like image and speech recognition.
- **Key Terminologies: Features, Labels, Training, and Testing Data**
    - **Features**: Input variables used to make predictions (e.g., age, income).
    - **Labels**: Output variables to be predicted (e.g., spam or not spam).
    - **Training Data**: Dataset used to train the model.
    - **Testing Data**: Dataset used to evaluate the model's performance.
    - **Validation Data**: Dataset used to tune hyperparameters and prevent overfitting.

Popular ML Algorithms (Overview)

- **Linear Regression**
    - Definition: A supervised learning algorithm for predicting continuous values.
    - Equation: y=mx+b$y=mx+b$, where yy is the dependent variable, xx is the independent variable, mm is the slope, and bb is the intercept.
    - Use cases: Predicting house prices, stock prices, etc.
- **Decision Trees**
    - Definition: A tree-like model for decision-making based on feature values.
    - Concepts: Nodes (decisions), branches (outcomes), and leaves (final predictions).
    - Use cases: Classification (e.g., loan approval) and regression.
- **K-Means Clustering**
    - Definition: An unsupervised learning algorithm for grouping data into clusters.
    - Concepts: Centroids, Euclidean distance, and convergence.
    - Use cases: Customer segmentation, image compression.

Key Use Cases of ML

- **Image Recognition**
    - Applications: Facial recognition, object detection, medical imaging.
    - Algorithms: Convolutional Neural Networks (CNNs).
- **Speech Recognition**
    - Applications: Virtual assistants (e.g., Siri, Alexa), transcription services.
    - Algorithms: Recurrent Neural Networks (RNNs), Transformers.
- **Recommendation Systems**

  - Applications: Netflix, Amazon, Spotify.
  - Algorithms: Collaborative filtering, matrix factorization.

Unit 3: Key Concepts in AI and ML - 9 hours

Data: The Foundation of AI and ML

- **Importance of Data in AI/ML Systems**
  - Data as the backbone of AI/ML: Why quality data is critical for model performance.
  - Data preprocessing: Cleaning, normalization, and transformation.
  - Data splitting: Training, validation, and test sets.
  - Data augmentation techniques for improving model robustness.
  - Challenges: Missing data, outliers, and imbalanced datasets.
- **Types of Data: Structured, Unstructured, and Semi-Structured**
  - **Structured Data**: Tabular data, relational databases, and CSV files.
  - **Unstructured Data**: Text, images, audio, and video.
  - **Semi-Structured Data**: JSON, XML, and NoSQL databases.
  - Data representation: Feature engineering, embeddings, and encoding (e.g., one-hot encoding, label encoding).

Algorithms: The Driving Force

- **Understanding the Role of Algorithms in ML Models**
  - What are algorithms? Mathematical frameworks for solving problems.
  - Types of ML algorithms: Supervised, unsupervised, and reinforcement learning.
  - Key concepts: Loss functions, optimization, and gradient descent.
  - Overfitting vs. underfitting: Causes and solutions (e.g., regularization, dropout).
- **Concept of Model Training and Optimization**
  - Training process: Forward propagation, backpropagation, and weight updates.
  - Hyperparameters: Learning rate, batch size, and epochs.
  - Optimization techniques: Stochastic Gradient Descent (SGD), Adam, and RMSprop.
  - Evaluation metrics: Accuracy, precision, recall, F1-score, and ROC-AUC.

The AI/ML Development Lifecycle

- **Data Collection, Model Training, Model Evaluation, and Model Deployment**
  - Data collection: APIs, web scraping, and public datasets.
  - Model training: Iterative process of learning from data.
  - Model evaluation: Metrics for classification, regression, and clustering.
  - Model deployment: Exporting models (e.g., Pickle, ONNX), APIs, and cloud integration.
- **Introduction to Cross-Validation and Model Tuning**
  - Cross-validation: K-fold, stratified K-fold, and leave-one-out.
  - Hyperparameter tuning: Grid search, random search, and Bayesian optimization.
  - Model selection: Comparing multiple models for best performance.

Ethics in AI and ML

- **Ethical Considerations in AI: Bias, Transparency, and Accountability**
  - Bias in AI: Types (e.g., selection bias, confirmation bias) and sources (e.g., biased datasets).
  - Transparency: Explainable AI (XAI) and interpretability techniques (e.g., SHAP, LIME).
  - Accountability: Legal and ethical responsibilities of AI practitioners.
  - Case studies: Real-world examples of AI ethics violations.
- **Responsible AI Practices**
  - Fairness: Ensuring equitable outcomes for all groups.
  - Privacy: Data anonymization and GDPR compliance.

    - Sustainability: Reducing the environmental impact of AI systems.
    - Tools for ethical AI: AI Fairness 360, Fairlearn, and What-If Tool.

Unit 4: Key Tools and Platforms for AI and ML - 9 hours

Introduction to Popular AI/ML Libraries and Frameworks

- **Python for AI/ML**
    - Why Python? Easy-to-read syntax, extensive libraries, and community support.
    - Key Python libraries for AI/ML:
        - **NumPy:** Numerical computing with arrays and matrices.
        - **Pandas:** Data manipulation and analysis with DataFrames.
        - **Matplotlib:** Data visualization and plotting.
- **Machine Learning Libraries**
    - **Scikit-learn:** A versatile library for traditional ML algorithms (e.g., regression, classification, clustering).
    - **TensorFlow:** An open-source framework for deep learning and neural networks.
    - **Keras:** A high-level API built on TensorFlow for simplifying deep learning model development.

Data Preprocessing in Machine Learning

- **Normalization and Standardization**
    - Normalization: Scaling data to a range of [0, 1].
    - Standardization: Scaling data to have a mean of 0 and a standard deviation of 1.
- **Handling Missing Values**
    - Imputation: Replacing missing values with statistical measures (mean, median, mode).
    - Dropping: Removing rows or columns with missing values.
- **Encoding Categorical Variables**
    - Label Encoding: Converting categories into numerical labels.
    - One-Hot Encoding: Creating binary columns for each category.

Deep Learning Basics

- **Neural Networks**
    - Structure: Input layer, hidden layers, and output layer.
    - Activation Functions: ReLU, Sigmoid, and Softmax.
- **Training Process**
    - Forward Propagation: Passing data through the network.
    - Backpropagation: Adjusting weights using gradient descent.
- **Loss Functions and Optimizers**
    - Loss Functions: Mean Squared Error (MSE), Cross-Entropy.
    - Optimizers: SGD, Adam, RMSprop.

Natural Language Processing (NLP)

- **Text Preprocessing**
    - Tokenization: Splitting text into words or sentences.
    - Stemming and Lemmatization: Reducing words to their root forms.
- **Feature Extraction**
    - Bag of Words (BoW): Representing text as word frequency vectors.
    - TF-IDF: Weighing words based on their importance in a document.
- **Sentiment Analysis Techniques**
    - Rule-based: Using predefined rules and lexicons.
    - Machine Learning-based: Training models on labeled datasets.

Introduction to Data Visualization

- **Importance of Visualizing Data for AI/ML**
    - Helps in understanding data distributions, patterns, and outliers.
    - Essential for exploratory data analysis (EDA) and model evaluation.
- **Overview of Data Visualization Tools**
    - **Matplotlib:** Basic plotting library for creating static, animated, and interactive visualizations.
    - **Seaborn:** Built on Matplotlib, provides high-level interfaces for statistical graphics.

Jupyter Notebooks for AI/ML Development

- **Setting Up Jupyter for ML Projects**
    - Installation and launching Jupyter Notebook.
    - Overview of the Jupyter interface (cells, kernels, and outputs).
- **Writing and Running Python Code for ML**
    - Using Jupyter for interactive coding, data exploration, and visualization.
    - Best practices for organizing ML projects in Jupyter Notebooks.

Unit 5: Future Trends in AI & ML - 10 hours

Convolutional Neural Networks (CNNs) and Image Classification

- **Fundamentals of CNNs**: Convolutional layers, pooling operations, activation functions, and fully connected layers.
- **Preprocessing Techniques**: Data normalization, augmentation, and dataset splitting strategies.
- **Performance Evaluation**: Metrics like accuracy, precision, recall, and F1-score.

Emerging AI Trends and Their Impact

- **AI in Robotics**: AI-driven robotics, reinforcement learning, and real-world industrial applications.
- **Natural Language Processing (NLP)**: Evolution from traditional models to transformers (GPT, BERT), applications in sentiment analysis, chatbots, and translation.
- **AI in Healthcare**: AI for medical imaging, disease prediction, drug discovery, and ethical concerns (bias, privacy, accountability).
- **AI in Automation**: Impact on industries like finance, retail, and manufacturing, along with job displacement concerns.

AI for Social Good

- **Tackling Global Challenges**:
    - **Climate Change**: AI in renewable energy, carbon footprint reduction, and environmental modeling.
    - **Healthcare**: AI in pandemic prediction, vaccine development, and medical accessibility.
    - **Education**: AI-driven personalized learning and tutoring systems.
- **Real-Time AI Applications**:
    - Chatbots: Intent recognition and response generation.
    - Recommendation Systems: Collaborative filtering and hybrid models.
    - Speech Recognition: Acoustic and language modeling.

AI in the Future: Opportunities and Challenges

- **Future AI Applications**:
    - **IoT & Smart Cities**: AI in traffic management, urban planning, and smart automation.
    - **Autonomous Systems**: AI in self-driving cars, drones, and logistics.
- **Challenges & Ethical Dilemmas**:
    - **Data Privacy & Security**: Risks of breaches and adversarial attacks.
    - **Bias & Transparency**: Algorithmic fairness and accountability.

Course Outcomes:

- Students will be able to define AI and differentiate between AI, Machine Learning, and Deep Learning.
- Students will be able to explain the key principles and components of AI systems.

- Students will be able to describe the different types of machine learning and their applications.
- Students will be able to apply Python and popular AI/ML libraries to perform basic machine learning tasks.
- Students will be able to analyze and discuss the ethical implications of AI technologies.
- Students will be able to identify and explain emerging trends in AI and their potential impact.

Skills Gained:

- Understanding of AI and ML concepts.
- Knowledge of machine learning algorithms.
- Data preprocessing and handling skills.
- Proficiency in Python and AI/ML libraries (NumPy, Pandas, Scikit-learn, TensorFlow, Keras).
- Ability to apply AI and ML techniques to real-world problems.
- Awareness of ethical considerations in AI.

Course Code	25EE01TP0204			
Category	Engineering Science Course			
Course Title	Computer Architecture and Organization			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	II

Course Outcomes:
Upon completion of this course, students will demonstrate the ability to:

1. **Understand** the common principles of computer architecture, multiprocessing and classify RISC/CISC.
2. **Develop** RISC-V assembly programs by understanding instruction formats
3. **Design** the RISC -V processor datapath and control on FPGA
4. **Apply** the concept of cache and virtual memory management in computer system.
5. **Evaluate** pipelined processor performance, identify hazards, and propose solutions to improve efficiency

Syllabus:

Module-I:

Introduction to computer system and its sub modules, Introduction to RISC and CISC paradigm, Functional Modeling of digital blocks.

Module-II:

Performance Equation, Common Principles of Computer organization: Amdahl's Law, Principle of Locality. Processor organization, instruction set (RISC-V), instruction formats, Representing Instructions in the Computer, Translating and Starting a Program, IEEE 754 floating point format.

Module-III:

Processor Design-Introduction, Datapath and control unit design, Performance Considerations, Multi-cycle design, Micro Programmed control design, Exception Handling.

Module-IV:

Motivation for Pipelining, Clock period and CPI, Pipelined data path, graphical representation, Pipelining Hazards.

Module-V:

Memory organization, concepts of semiconductor memory, memory management, concept of cache and associative memories, virtual memory.

Module-VI:

Parallel processing concepts, multiprocessors and its characteristics, Input/output Subsystem: -Interfaces and buses, I/O Operations, Designing I/O Systems, Overview of Domain-Specific Architectures

Text Books:

1. Computer Organization and Design RISC-V Edition - The Hardware/Software Interface, David A. Patterson, John L. Hennessy, 2nd Edition, 2021.

Reference Books:

1. Computer Architecture and Organization; J. P. Hayes; Third Edition (Fifth Reprint), McGraw Hill, 2012.
2. Computer Architecture And Parallel Processing; Kai Hawang, Faye A. Briggs, McGraw Hill, 2012

List of Experiments:

1. To design and integrate ALU and ALU control unit of RISC-V processor together and test it using a test bench. Encode the instructions to be executed by ALU and generate these encoded test vectors to control the ALU operations. Supply random test inputs from test bench to check the results using Vivado and FPGA.

2. Integrate register Files, ALU control, ALU together. Write a top module to connect these modules using wires in Verilog HDL. Demonstrate any one type of RISC-V instruction. Ensure the register files are getting initialized with the correct data and encoded instructions. Uses test bench to generate the relevant test vectors.

3. Write a top module to connect the instruction memory, data memory, register files, ALU and ALU control together. Encode one logical and one arithmetic instruction in the instruction memory. Initialize the data memory fields with the data. Verify the instructions fetch process from instruction memory. Write a Verilog test bench to verify arithmetic and logical instructions by applying suitable test cases and control signals.

4. Write a control path in Verilog HDL to control the execution of the data path of MIPS instructions in experiment3. Analyze the control signals for S type, R type and I type instructions using a suitable test bench.

5. Encode the assembly program in instruction memory with at least 1 R-type, 1 immediate type and 1 branch-type instruction in the memory. Demonstrate the execution of this program using Vivado tool and FPGA.

Course Code	25HS02TP0201
Category	Ability Enhancement Course

Course Title	English for Professional Communication			
Scheme & Credits	L	P	Credits	Semester
	2	2	3	II

Course Outcomes

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

1. Demonstrate effective use of word power in written as well as oral communication.
2. Understand the techniques of listening and apply the techniques of reading comprehension used in professional communication.
3. Apply the principles of functional grammar in everyday as well as professional communication.
4. Effectively implement the comprehensive principles of written communication by applying various writing styles.
5. Create precise and accurate written communication products.

Syllabus

Module I: Vocabulary Building

- Importance of using appropriate vocabulary
- Techniques of vocabulary development
- Commonly used power verbs, power adjectives and power adverbs
- Synonyms, antonyms, phrases & idioms, one-word substitutions, and standard abbreviations

Module II: Listening and Reading Comprehension

- **Listening Comprehension:** Active listening, reasons for poor listening, traits of a good listener, and barriers to effective listening
- **Reading Comprehension:** Types and strategies

Module III: Functional Grammar and Usage

- Identifying common errors in use of: articles, prepositions, modifiers, modal auxiliaries, redundancies, and clichés
- Tenses
- Subject-verb agreement, noun-pronoun agreement
- Voice

Module IV: Writing Skills

- Sentence Structures
- Sentence Types
- Paragraph Writing: Principles, Techniques, and Styles

Module V: Writing Practices

- Art of Condensation: Précis and Note Making
- Correspondence writing techniques and etiquettes – academic writing
- Group discussion, Résumé writing

Reference Books

1. *Communication Skills* – Sanjay Kumar and Pushp Lata, Oxford University Press, 2011
2. *Practical English Usage* – Michael Swan, Oxford University Press, 1995
3. *Remedial English Grammar* – F. T. Wood, Macmillan, 2007
4. *On Writing Well* – William Zinsser, Harper Resource Book, 2001
5. *Study Writing* – Liz Hamp-Lyons and Ben Heasly, Cambridge University Press, 2006
6. *Exercises in Spoken English* (Parts I–II), CIEFL, Hyderabad, Oxford University Press

English for Professional Communication Lab

Course Objective
To enhance competency of communication in English among learners

Course Outcomes
On completion of English Lab course, students will be able to achieve the following:
CO1: Apply effective speaking skills
CO2: Apply effective listening and reading skills
CO3: Demonstrate the techniques of effective public speaking and presentation skills
CO4: Evaluate and apply the effective strategies for Group Discussions

Syllabus
List of practicals

Computer Assisted + Activity Based Language Learning

Practical 1: Everyday Situations: Conversations and Dialogues – Speaking Skills
Practical 2: Pronunciation, Intonation, Stress, and Rhythm
Practical 3: Everyday Situations: Conversations and Dialogues – Listening Skills
Practical 4: Reading Skills

Activity Based Language Learning

Practical 5: Public Speaking
Practical 6: Presentation Skills: Orientation
Practical 7: Presentation Skills: Mock
Practical 8: Group Discussions: Practice
Practical 9: Group Discussions: Mock

Reference Books

1. *Communication Skills*. Sanjay Kumar and Pushp Lata. Oxford University Press. 2011.
2. *Practical English Usage.* Michael Swan. OUP. 1995.

Course Code	25HS02TH0203			
Category	Indian Knowledge System			
Course Title	Foundational Literature of Indian Civilization			
Scheme & Credits	L	P	Credits	Semester
	1	0	1	II

Course Outcomes

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

1. Understand the Indian knowledge system and its scientific approach.
2. Get introduced to the Vedic corpus and recognize the multi-faceted nature of the knowledge contained in the Vedic corpus.
3. Understand the salient features of the philosophical systems of the Vedic and non-Vedic schools.
4. Develop a basic understanding of the ancient wisdom recorded in various Indian literary works.

Syllabus

Module I: Overview of Indian Knowledge System

- Importance of ancient knowledge
- Defining IKS
- IKS classification framework
- Historicity of IKS
- Some unique aspects of IKS

Module II: The Vedic Corpus

- Introduction to Vedas
- Four Vedas and their divisions
- Six Vedangas
- Distinct features of Vedic life

Module III: Indian Philosophical Systems

- Development and unique features
- Vedic schools of philosophy:
    - Samkhya and Yoga School
    - Nyaya and Vaisheshika School
    - Purva-Mimamsa and Vedanta schools of philosophy
- Non-Vedic philosophies:
    - Jainism, Buddhism, and other approaches

Module IV: Vedic Maths -1

- Introduction of Vedic Mathematics
- Bodhyan geometry
- circular functions
- inverse circular functions

Module V: Vedic Maths -2

- Multiplication of polynomials using:
  - Nikhilaṃ Sutra
  - Ūrdhvatiryagbhyām Sutra
  - Verification using Gunitasamuccayaḥ Sutra
- Division of two polynomials using:
  - Parāvartya Yojayet Sutra
- HCF and LCM of two polynomials using:
  - Ādyamādyenāntyamantyena Sutra
  - Ānurūpyeṇa Sutra
- Factorization of polynomials up to degree 3 using:
  - Ānurūpyeṇa Sutra
  - Lopanasthāpanābhyām Sutra
  - Ādyamādyenāntyamantyena Sutra

Reference Material

1. B. Mahadevan, Vinayak Rajat Bhar, Nagendra Pavana R. N., *Introduction to Indian Knowledge System: Concepts and Applications*, PHI, 2022
2. S. C. Chatterjee and D. M. Datta, *An Introduction to Indian Philosophy*, University of Calcutta, 1984

Course Code	25HS04PR0201			
Category	Co-Curricular Activity			
Course Title	Health-Fitness-Wellbeing (HFW)			
Scheme & Credits	L	P	Credit	Semester
	0	2	1	II

Aim of the Course: The course aims to foster Health and wellness through Healthy and Active Lifestyle and creating awareness about the fundamentals of Physical Education, Sports, Yoga, Recreation and its effectiveness through practical experiences and hands on activities.

Objectives of the Course:

1. To impart the students with basic concepts of Sports, Yoga and Recreational activities for health and wellness.
2. To familiarize the students with health-related Exercise and evaluate their Health-related Fitness.
3. To make Overall growth & development with team spirit, social values and leadership qualities among students through various sports, games and Yogic activities.
4. To create Environment for better interaction and recreation among students as neutralizer for stress through various minor and recreational games.

Course Outcomes: On completion of the course, students will be able to:

1. Understand fundamental skills, basic principle and practices of sports and Yoga.
2. Practically learn the principles of implementing general and specific conditioning of physical exercises and yoga.
3. Develop Health-related fitness and Body-mind co-ordination through various fitness activities, sports, recreational games and yoga.
4. Practice Healthy & active living with reducing Sedentary Life style.

Course Content:

Module 1:

- Warm up and Cool Down and Stretching Exercises.
- General and Specific Exercises.
- Calculation of BMI & Resting Pulse Rate.
- General and Specific exercises for strength, Speed, Agility, Cardiovascular Endurance, Flexibility, Coordinative abilities.
- Practice of Fundamental Skills of Volleyball, Table Tennis and Chess, etc.
- Knowledge and practice of the Equipment used in a Gymnasium and its application.

Module 2:

- Yoga: Standing, Sitting, Prone & Supine positions.
- Suryanamaskar.
- Pranayama, Meditation and Relaxation Techniques.
- Recreational Games.
- Practice of Fundamental Skills of Basketball, Football, Carrom, etc.
- Health related Physical Fitness Test.

Assessment Pattern:

Assessment Type	Weightage in Marks	Total Marks
Practical	Physical Efficiency Test – 30 Marks Sports/Games skill Activity/Project – 10 Marks Yoga Activities – 10 Marks	50
		Total – 50 Marks

References:

1. Russell, R.P. (1994). Health and Fitness Through Physical Education. USA: Human Kinetics.
2. Uppal, A.K. (1992). Physical Fitness. New Delhi: Friends Publication.
3. AAPHERD "Health related Physical Fitness Test Manual."1980 Published by Association drive Reston Virginia
4. Kumar, Ajith. (1984) Yoga Pravesha. Bengaluru: RashtrothannaPrakashana.
5. Dr. Devinder K. Kansal, A Textbook of Test Evaluation, Accreditation, Measurements and Standards (TEAMS 'Science)

SYLLABUS OF SEMESTER III

Course Code	25EE01TP0301-1 / 25EE01TP0301-2			
Category	Programme Core Course			
Course Title	Object Oriented Programming using Python OR Object Oriented Programming using JAVA			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	III

Course Outcomes

On successful completion of the course, students will be able to

1. Understand the principles of object-oriented programming; create classes, instantiate objects and invoke methods.
2. Apply the concepts of generics and implement collection classes and develop reusable programs using the concepts of OOP.
3. Apply the concepts of Multithreading and Exception handling to develop efficient and error free Codes for solving classic synchronization problems.
4. Utilize modern design tools and collection framework to solve real world problems

Syllabus

Module I
Features of Object Oriented Programming languages, Abstraction, Encapsulation, Inheritance, polymorphism and late binding. Concept of a class, Access control of members of a class, instantiating a class, constructor and method overloading and overriding.

Module II
Concept of inheritance, methods of derivation, use of super keyword and final keyword in inheritance, run time polymorphism, abstract classes and methods, Interface, implementation of interface.

Module III
Creating packages, importing packages, static and non-static members, Lambda Expressions Introduction, Block, Passing Lambda expression as Argument.

Module IV
Exceptions, types of exception, use of try catch block, handling multiple exceptions, using finally, throw and throws clause, user defined exceptions, file handling in Java, Serialization.

Module V
Generics, generic class with two type parameter, bounded generics. Collection classes: Array list, Linked List, Hash set, Tree set, Hash Map.

Module VI
Multithreading: Java Thread models, creating thread using runnable interface and extending Thread,

thread priorities, Thread Synchronization, Inter Thread communications. Introduction to Design Patterns, Need of Design Pattern, Classification of Design Patterns.

Text Books

1. Herbert Schildt; JAVA, the Complete Reference; Ninth Edition, Tata McGraw- Hill Publishing Company Limited.
2. Design Patterns by Erich Gamma, Pearson Education.

LIST OF EXPERIMENTS

1.Implement the Classes and Objects in Java.

2. Implement a program in java with Constructors and destructors. Also implement the concept of overloading.

3. Demonstrate use of Inheritance.

4. Implement a concept of Interface in Java.

5.Demonstrate use of Multi Thread.

6.Implement Packages and import that package in program.

7.Demonstrate use of Exception handling mechanism.

8.Apply concept of generics class and method.

9.Demonstrate collection framework and perform some basic operations on the Array List and

Hashset.

10.Apply file handling concepts in Java.

Course Code	25EE01TP0302			
Category	Program Core Course			
Course Title	IoT Fundamentals and Architecture			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	III

Course Outcomes

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

1. Understand the basics of networking
2. Gain the knowledge about iot standards
3. Realize the basic applications using arduino and raspberry pi
4. Illustrate different real world applications syllabus

Syllabus

Module - I: (7Hrs)

Basics of Networks, TCP/IP model,IP Addresses, application layer protocols, HTTP, MQTT, WWW, constraint application protocol,stacks.

Module - II: (7Hrs)

Introduction to IoT, evolution of IoT, IoT and SCADA, Big Data, IoT Standards, requirement, Platforms, relevance of IoT, security

Module - III: (7Hrs)

Interoperability in IoT, Machine-to-Machine Communications, Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, Sensing, Actuation, Sensor Networks

Module - IV: (7Hrs)

Introduction to Python programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi.

Module - V: (7Hrs)

Introduction to SDN, Fog Computing, IoT application case studies: Smart Cities and Smart Homes, Connected Vehicles, Smart Grid, Industrial IoT, Agriculture, Healthcare, Activity Monitoring, IoT in India: Smart India projects, Challenges in IoT

Text Books

1. Computer Networks: A Top-Down Approach; Behrouz A Forouzan, Firouz Mosharraf, McGraw Hill Education. Special Indian Edition 2012
2. Arduino Cookbookby Michael Margolis, O'Reilly Media, Inc., 1st edition
3. Raspberry pi Cookbook by Simon Monk, O'Reilly Media, Inc., 3rd edition.

Course Code	25EE01TP0303			
Category	Programme Core Course			
Course Title	Machine Learning Engineering			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	III

Course Outcomes:

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

1. Perform exploratory data analysis to prepare datasets for machine learning models.
2. Employ calculus, linear algebra, probability theory, and optimization methods to develop machine-learning models.
3. Implement and analyze supervised and unsupervised learning algorithms on a given data set.
4. Evaluate and interpret the performance of machine learning models using evaluation metrics and inference techniques to derive meaningful insights.
5. Apply appropriate machine learning techniques to solve real-world problems by selecting suitable models, algorithms, and optimization strategies.

Syllabus:

Module 1: (6 Hrs)

Foundations for ML: Review of Linear algebra and Optimization, introduction to machine learning and its types, parametric vs non-parametric models, Machine Learning pipeline, Exploratory Data Analysis.

Module 2: (8 Hrs)

Supervised learning algorithms: Linear and Logistic Regression – Bias/Variance Trade-off, overfitting and under fitting, Regularization, Multivariate and polynomial Regression, Variants of Gradient Descent algorithm. Decision Trees, Basic decision trees learning algorithm, Bagging, Boosting, and Random Forests.

Module 3: (8 Hrs)

Support Vector Machines, and Kernel functions in SVM, K-Nearest Neighbors. Feature selection techniques, Feature scaling, Evaluation and Model Selection: Performance Metrics: Accuracy, Precision, Recall, F1-score, Confusion Matrix, ROC & AUC Curves, Evaluation Measures, Cross-Validation techniques.

Module 4: (6 Hrs)

Probabilistic Machine Learning: Bayesian learning and Bayesian networks, Naive Bayes classifier; Bayes optimal classifiers, Maximum Likelihood Estimation, MAP; Gaussian Discriminant Analysis.

Module 5: (7 Hrs)

Unsupervised learning algorithms: K-means clustering, Hierarchical Clustering, DBSCAN, Anomaly Detection: Isolation Forests, One-Class SVM., Dimensionality Reduction techniques: PCA, LDA; Anomaly detection, Recommender System.

Text Book:

1. Machine learning, by Mitchell Tom, First edition, McGraw Hill, 1997.
2. The Elements of Statistical Learning Data Mining, Inference, and Prediction by Trevor Hastie, Robert Tibshirani, Jerome Friedman, Second Edition, Springer, 2009.

Reference Books:

1. Pattern Recognition and Machine Learning by Christopher M. Bishop, First edition, Springer, 2006.
2. Machine Learning: A Probabilistic Perspective by Kevin P. Murphy, Francis Bach; MIT Press, 2012.
3. Understanding Machine Learning: From Theory to Algorithms by Shai Shale-Shwartz, and Shai Ben-David, Cambridge University Press, 2014.

List of Experiments:

Lab-01: **Implement data preprocessing techniques on the given dataset.**

a. Perform Exploratory Data Analysis (EDA)
b. Decide on strategies for handling missing data (e.g., imputation, deletion, interpolation).
c. Identify and remove duplicate entries from the dataset if any.
d. Detect outliers and decide on appropriate treatment methods (e.g., removal, transformation, binning).
e. Convert categorical variables into numerical representations suitable for machine learning algorithms (E.g. one-hot encoding, label encoding, and target encoding).
f. Standardize or normalize numerical features to ensure they have a similar scale, preventing certain features from dominating the learning process.
g. Create new features from existing ones or transform existing features to improve model performance (E.g. polynomial features, interaction terms, or domain-specific transformations)
h. Use techniques like filter methods (e.g., correlation analysis), wrapper methods (e.g., recursive feature elimination) for feature selection.
i. Divide the dataset into training and testing sets to evaluate the performance of the machine-learning model.
j. Visualize the dataset to gain insights into its distribution, relationships between features, and potential patterns.
k. Explore summary statistics, histograms, scatter plots, and correlation matrices to understand the data's characteristics and inform preprocessing decisions.

Lab-02: Implement linear regression algorithm (Single, Multiple variable and polynomial) using benchmark datasets and evaluate the performance of linear regression using evaluation measures like MAE, MSE, RMSE, Coefficient of Determination (R^2), and **Adjusted R-squared.**

Lab-03: Implement **the following algorithms to perform the task of classification on the** benchmark datasets and evaluate the performance of algorithms using evaluation measures like Accuracy, Precision, Recall, F1 score, ROC curves, AUC, and cross-validation techniques.

a. Logistic Regression
b. Decision Tree
c. Random Forest
d. K-nearest Neighbor

Lab-04: Build and implement **an image classifier using Support Vector machine (SVM) algorithm** and evaluate the performance of the trained model algorithms using k-fold cross-validation.

Lab-05: Build and develop a model for document classification using probabilistic machine learning algorithms.

Lab-06: Implement the K-means clustering algorithm to perform image segmentation and compare its performance with different numbers of clusters (k) using various evaluation metrics such as silhouette score, Davies-Bouldin index, and within-cluster sum of squares (WCSS).

Lab-07: Perform Dimensionality Reduction using Principal Components Analysis (PCA) and do the following task:

a. Use PCA in order to **visualize** a high-dimensional problem in 2-dimensions.
b. Use PCA in order to **improve model-training time** and understand the **speed-accuracy trade-off.**
c. Evaluate the trade-offs between preserving global structure and local relationships in the data space.
d. Discuss when to use PCA and when not to use it.

Lab-08: Implement **Gaussian Mixture Model (GMM) Clustering** to model complex data distributions and visualize the resulting cluster assignments and probability contours.

Lab-09: Investigate the effectiveness of Isolation Forest for identifying outliers and to detect anomalous behavior in server computers.

Lab-10: Build a simple recommender system using collaborative filtering or matrix factorization techniques and assess its performance using metrics like Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE).

Lab-11: Perform the comparative analysis of ensemble learning techniques on classification tasks.

Lab-12: **A Capstone Project:** Students are required to utilize the knowledge and competencies gained throughout the course to address a practical real-world challenge or investigate a substantial research query within the realm of machine learning.

Course Code	25EE01TP0304			
Category	Programme Core Course			
Course Title	Digital Signal Processing			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	III

Course Outcomes:
At the end of this course, students will demonstrate the ability to

1. Describe discrete-time signals in different forms and analyze the behavior of Linear Time-Invariant (LTI) systems in the frequency domain.

2. Apply Z-transform techniques to process signals in various discrete-time systems, and interpret their system behavior.

3. Design and implement Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, and evaluate their performance to ensure they meet expected system specifications.

4. Analyze finite word-length effects due to rounding and truncation, examine the impact of multi-rate signal processing, and understand the fundamentals of DSP processors.

Syllabus:

Module I (5 Hrs):
Discrete Time Fourier Transform (DTFT): Analysis of LTI system using DTFT, block diagram and signal flow graph representation of linear constant coefficient difference equations.

Module II (7 Hrs):
Z-transform: Z-transform and its properties, analysis of LTI discrete time system using Z transform, Relation between Laplace and Z transform, Inverse Z-transform, Unilateral Z- transform.

Module-III (7 Hrs):
Discrete Fourier Transform (DFT): Frequency Domain sampling, DFT and its properties, filtering of long data sequences using overlap-save method and overlap-add method, Radix-2 Fast Fourier Transform (FFT) algorithms.

Module IV (08 Hrs):
Design of FIR filter: Digital filter concepts, FIR filters Design techniques: Fourier series, Windows (Rectangular, Bartlett, Hanning, Hamming, Blackman, Kaiser) and Optimal frequency sampling, structures for FIR systems

Module V (5 Hrs):

Design of IIR filter: Impulse invariance transformation, Bilinear Transformation, Design of Butterworth and Chebyshev filters, structures for IIR systems.

Module VI (5 Hrs):

DSP hardware, Finite word length effects, Multirate signal processing: Quantization by truncation and Rounding, Quantization of Input data and filter coefficients, Digital Signal Processing applications, introduction to DSP processors, multirate signal processing.

Text Book:

1. Digital Signal Processing: Principles, Algorithms & Applications, John G. Proakis & Dimitris G. Manolakis, PHI, 4th Edition

Reference Book:

1. Digital Signal Processing: AComputerbased Approach, Sanjit K. Mitra, 4th Edition Mc-GrawHill.
2. Discrete Time Signal Processing, Alan V. Oppenheim & Ronald W. Schafer, 3rdEdition, Pearson.
3. Digital Signal Processing, Thomas J. Cavicchi, Wiley Publication, Student Edition.
4. Digital Signal Processing, A NagoorKani, 2nd Edition Mc-Graw Hill.

Course Outcomes:

At the end of this course, students will demonstrate the ability to

1. Describe discrete-time signals in different forms and analyze the behavior of Linear Time-Invariant (LTI) systems in the frequency domain.

2. Apply Z-transform techniques to process signals in various discrete-time systems, and interpret their system behavior.

3. Design and implement Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, and evaluate their performance to ensure they meet expected system specifications.

4. Analyze finite word-length effects due to rounding and truncation, examine the impact of multi-rate signal processing, and understand the fundamentals of DSP processors.

Syllabus:

Experiments based on **24EE01TH0304 (Digital Signal Processing)** Syllabus.

Course Code	25EE01TH0305			
Category	Multidisciplinary Minor -1			
Course Title	Introduction to IoT system Design			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	III

Course Outcomes:

1. Understand the IoT reference Model and different element used in it.
2. Illustrate Python Programming for IoT
3. Analyze the role of different elements in IoT as a system in different use cases.
4. Propose a solution to a real-world problem using the IoT framework
5. Evaluate the engineering feasibility of the solutions'/ Use cases.

Syllabus:

Module 1:

Introduction to Internet of Everything, IoT Reference Model, Different IoT models, Elements in IoT Infrastructure

Module 2:

IoT Infrastructure Elements and their roles at Different Layer in IoT Reference Model, Devices/ Function of elements in IoT Sensors, Controllers, Network, Cloud, User Applications and Data Analytics

Module 3:

Logical Design using Python- Introduction, Installing Python, Python Data Types and Data structures, Control flow, Functions, Modules, File Handling, Operations, Classes, Python Packages of Interest for IoT.

Module 4:

Perception Layer, Network Layer, Application Layer Architecture in IoT system, Resources used at Perception Layer, Network Layer, Application Layer.

Module 5:

Use cases of IoT Systems builds across Raspberry Pi/ Arduino Uno/Nano IOT/ ESP32 / Node MCU/ PI-PICO H/W variants etc,

Text Book:

1.Internet of Things Principles and Paradigms, Rajkumar Buyya and Amir Vahid Dastjerdi, Morgan Kaufman, Elsevier 2016 1st Edition
2.Python Programming using Problem Solving Approach, Reema Thareja, 2017,
OXFORD University Press

Reference Books:

1.Internet of Things Principles, Paradigms and Application of IoT, Joseph Kofi Wireko,
2.Kaml Hiran, BPB Publications 2020 1st Edition
3.Arduino NanoBLE/ Nano 33IoT Application notes
4.Espressif Application notes ESP32/ESP8266/Node MCU Raspberry Pi- PICO application notes

Course Code	25EEOEC01TH0306			
Category	Open Elective			
Course Title	Open Elective-I/MOOCs/Industry Offered			
Scheme & Credits	L	P	Credits	Semester
	2	0	2	III

Note: MOOC Courses from Coursera platform

Course Code	25HS02TH0301				
Category	Vocational Enhancement Course				
Course Title	Foundation Course In Universal Human Value				
Scheme & Credits	L	T	P	Credits	Semester
	1	0	0	1	III

Course Outcomes

On successful completion of the course, students will be able to

1. Develop a holistic perspective of life
2. Better understanding of inter-personal relationships and relationship with society and nature.
3. An ability to strengthen self-reflection

Syllabus

Module 1: Aspirations and concerns

Need for Value Education: Guidelines and content of value education.
Exploring our aspirations and concerns: Knowing yourself, Basic human aspirations
Need for a holistic perspective, Role of UHV;
Self-Management: harmony in human being

Module 2: Health

Harmony of the Self and Body, Mental and physical health;
Health for family, friends and society.

Module 3: Relationships and Society

Harmony in relationships, Foundational values: Trust, Respect, Reverence for excellence, Gratitude and love;
harmony in society; harmony with nature.

Reference Material

The primary resource material for teaching this course consists of

1. Textbook: R.R. Gaur, R. Sangal, G. P. Bagaria, *A foundation course in Human Values and professional Ethics*, Excel books, New Delhi, 2010, ISBN 978-8-174-46781-2

Reference books

1. B. L. Bajpai, 2004, *Indian Ethos and Modern Management*, New Royal Book Co., Lucknow. Reprinted 2008.
2. P. L. Dhar, R. R. Gaur, 1990, *Science and Humanism*, Commonwealth Publishers.
3. Sussan George, 1976, *How the Other Half Dies*, Penguin Press. Reprinted 1986, 1991
4. Ivan Illich, 1974, *Energy & Equity*, The Trinity Press, Worcester, and HarperCollins, USA
5. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, *Limits to Growth*, Club of Rome’s Report, Universe Books.
6. Subhas Palekar, 2000, *How to practice Natural Farming*, Pracheen (Vaidik) Krishi Tantra Shodh, Amravati.
7. A. Nagraj, 1998, *Jeevan Vidya ek Parichay*, Divya Path Sansthan, Amarkantak.
8. E. F. Schumacher, 1973, *Small is Beautiful: a study of economics as if people mattered*, Blond & Briggs, Britain.A. N. Tripathy, 2003, *Human Values*, New Age International Publishers.

Course Code	25EE01HT0301			
Course Type	Honors Specialization in Data Science			
Course Title	Fundamentals of Artificial intelligence and Edge Computing			
Scheme& Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes:

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

1. Comprehend the fundamental concepts of Artificial Intelligence (AI), including machine learning, deep learning, and neural networks.
2. Implement neural network architectures using popular deep learning frameworks for tasks such as image recognition or natural language processing.
3. Recognize the significance of edge computing in the context of the Internet of Things (IoT) and explain the architecture of edge computing systems.
4. Analyze real-world use cases of AI at the edge and evaluate the benefits and limitations of edge AI frameworks.
5. Develop innovative strategies for deploying AI algorithms on resource-constrained edge devices while minimizing latency.

Syllabus

Module 1: Fundamentals of Artificial Intelligence: (05 Hours)

Introduction to Artificial Intelligence (AI), history, and evolution of AI, basic concepts including machine learning, deep learning, and neural networks.

Module 2: Machine Learning Algorithms: (07 Hours)

Overview of machine learning algorithms, supervised and unsupervised learning, regression, classification, clustering, and reinforcement learning.

Module 3: Deep Learning and Neural Networks: (07 Hours)

Introduction to deep learning, neural network architectures, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and their applications.

Module 4: Edge Computing and IoT: (07 Hours)

Introduction to edge computing, Systems Paradigms of Edge computing, Edge Computing Frameworks, Value Scenarios for Edge Computing. Edge computing system architectures. Industrial Applications of Edge Computing, Intelligent Edge and Edge Intelligence, Challenges and opportunities in Edge Computing. its importance in IoT (Internet of Things).

Module 5: AI at the Edge: (07 Hours)

Artificial Intelligence Training at Edge : Distributed Training at Edge, Federated Learning(FL) at Edge, Communication-Efficient FL, Resource-Optimized FL, Security-Enhanced FL Case studies based on training at edge Artificial Intelligence Inference in Edge : Optimization of AI Models in Edge: General methods, Edge device Segmentations of AI models, Early Exit of Inference (EEoI) , Sharing of AI Computation

Module 6: Applications and Future Trends & Challenges: (07 Hours)

Applications of artificial intelligence at the edge: Real-time Video Analytic, Autonomous Internet of Vehicles (IoVs), Intelligent Manufacturing, Smart Home and City, Urban Healthcare, Urban Energy Management, Manufacturing, Transportation and traffic. real-world use cases, edge AI frameworks, and edge AI deployment strategies. Emerging trends in AI and edge computing, ethical considerations, privacy concerns, and future challenges and opportunities.

Text books:

1. "Artificial Intelligence: A Modern Approach," by Stuart Russell and Peter Norvig, Pearson, 4th edition, 2021.
2. "Deep Learning," by Ian Goodfellow, Yoshua Bengio, and Aaron Courville, the MIT Press, 1st edition, 2016.
3. "Edge AI: Convergence of Edge Computing and AI," by Xiaofei Wang, Yiwen Han, Victor C. M. Leung, Dusit Niyato, Xueqiang Yan, Xu Chen, Wiley, 1st edition, 2021.

Reference Books:

1. "Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig, Pearson, 4th edition, 2020.
2. "Artificial Unintelligence: How Computers Misunderstand the World" by Meredith Broussard, The MIT Press, 1st edition, 2018.
3. "The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power" by Shoshana Zuboff, PublicAffairs, 1st edition, 2019.
4. "Neural Networks and Deep Learning: A Textbook" by Charu C. Aggarwal, Springer, 1st edition, 2018.
5. "Edge Computing: A Primer" by Agustinus Borgy Waluyo and Huansheng Ning, Wiley, 1st edition, 2020.
6. "Edge Computing: Models, Technologies, and Applications" by Khaled Salah, Mohsen Guizani, et al., Wiley, 1st edition, 2021.
7. "Practical Deep Learning for Cloud, Mobile, and Edge" by Anirudh Koul, Siddha Ganju, and Meher Kasam, O’Reilly Media, 1st Edition: 1st, 2019.
8. "Learning Edge AI" by R. Stanley Williams, Wiley, 1st Edition: 1st, 2021.

Course Code	25EE10MT0301			
Category	Minor Specialization			
Course Title	IoT Fundamentals			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	III

Course Outcomes

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

1. Understand the basics of networking
2. Gain the knowledge about iot standards
3. Realize the basic applications using arduino and raspberry pi
4. Illustrate different real world applications syllabus

Syllabus

Module - I: (7Hrs)

Basics of Networks, TCP/IP model,IP Addresses, application layer protocols, HTTP, MQTT, WWW, constraint application protocol,stacks.

Module - II: (7Hrs)

Introduction to IoT, evolution of IoT, IoT and SCADA, Big Data, IoT Standards, requirement, Platforms, relevance of IoT, security

Module - III: (7Hrs)

Interoperability in IoT, Machine-to-Machine Communications, Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, Sensing, Actuation, Sensor Networks

Module - IV: (7Hrs)

Introduction to Python programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi.

Module - V: (7Hrs)

Introduction to SDN, Fog Computing, IoT application case studies: Smart Cities and Smart Homes, Connected Vehicles, Smart Grid, Industrial IoT, Agriculture, Healthcare, Activity Monitoring, IoT in India: Smart India projects, Challenges in IoT

Text Books

4. Computer Networks: A Top-Down Approach; Behrouz A Forouzan, Firouz Mosharraf, McGraw Hill Education. Special Indian Edition 2012
5. Arduino Cookbookby Michael Margolis, O'Reilly Media, Inc., 1st edition
6. Raspberry pi Cookbook by Simon Monk, O'Reilly Media, Inc., 3rd edition.

SYLLABUS OF SEMESTER IV

Course Code	25EE01TP0402-1			
Category	Program Elective Course			
Course Title	Natural Language Processing			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	IV

Course Outcomes

1. Understand the fundamental tasks in natural language processing (NLP) related to syntax, semantics, and pragmatics.
2. Apply knowledge of natural language annotation techniques and text analysis tools.
3. Understand and apply statistical parsing techniques.
4. Implement Semantic Role Labeling (SRL) and Semantic Parsing techniques for understanding sentence meaning and Information Extraction.
5. Identify issues and challenges in Machine Translation.

Syllabus

Module I: Introduction (6 Hours)
NLP tasks in syntax, semantics, and pragmatics. Key issues & Applications such as information extraction, question answering, and machine translation. The problem of ambiguity. The role of machine learning. Brief history of the field.

Module II: N-gram Language Models (6 Hours)
Role of language models. Simple N-gram models. Estimating parameters and smoothing. Evaluating language models. Part of Speech Tagging and Sequence Labeling Lexical syntax. Hidden Markov Models. Maximum Entropy models.

Module III: Syntactic Parsing (6 Hours)
Grammar formalisms and tree banks. Efficient parsing for context-free grammars (CFGs). Statistical parsing and probabilistic CFGs (PCFGs). Lexicalized PCFGs.

Module IV: Semantic Analysis (7 Hours)
Lexical semantics and word-sense disambiguation. Compositional semantics. Semantic Role Labeling and Semantic Parsing.

Module V: Information Extraction (IE) (5 Hours)
Named entity recognition and relation extraction. IE using sequence labeling. Automatic summarization. Subjectivity and sentiment analysis.

Module VI: Machine Translation (MT) (5 Hours)
Basic issues in MT. Statistical translation, word alignment, phrase-based translation, and synchronous grammars.

Textbook

1. D. Jurafsky and R. Martin, *Speech and Language Processing*, 2nd edition, Pearson Education, 2009.
2. Allen and James, *Natural Language Understanding*, Second Edition, Benjamin/Cumming, 1995.
3. Charniack & Eugene, *Statistical Language Learning*, MIT Press, 1993.

Reference Book

1. Akshar Bharati, Vineet Chaitanya, and Rajeev Sangal, *NLP: A Paninian Perspective*, Prentice Hall, New Delhi, 1994.
2. T. Winograd, *Language as a Cognitive Process*, Addison-Wesley, 1983.

Course Code	25EE01TP0402-3			
Category	Program Elective Course			
Course Title	Digital System Design			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	IV

Course Outcomes

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

1. Utilize/Apply the knowledge of digital circuits to design basic combinational and sequential blocks and hierarchical implementation of digital systems
2. Make use of dataflow, structural and behavioral modelling styles of verilog HDL for simulating the combinational/sequential circuits and systems
3. Understand, design and analyse the functionality of digital systems
4. Modelling of FSM, Data path and control unit.
5. Use of EDA tools and FPGA development platform for digital system design, verification, testing and implementation.

Syllabus

Module I (6 Hrs)

Digital System Design Flow, FPGA Architecture, Introduction to FPGA Development Board, Introduction to HDL, Basic Language Elements, Syntax and Semantics of HDL

Module II (8 Hrs)

Gate level, Dataflow and Behavioral Modeling for combinational circuits like Multiplexer, De-multiplexer, Encoder-Decoder, Flip-Flop, Counter, Writing Test Benches and Handling Text files to test the Circuits.

Module III (6 Hrs)

Design and Analysis of Standard Combinational Blocks, Algorithm to Architectural Translation for Arithmetic Circuits-Adders, Subtractor, Multiplier, Divider, Shifter, ALU and Comparator

Module IV (6 Hrs)

Design and analysis of standard sequential blocks, Finite State Machine Design.

Module V (6 Hrs)

Design of Data Path and Control unit with Case Studies.

Module VI (6 Hrs)

Logic Synthesis and Optimization Techniques for Area, Power and Delay, Timing analysis-Setup and Hold Violations, Synthesis of HDL code on FPGA platforms, Concepts of Critical Path Delay

Text Book

1. Verilog HDL: A Guide to Digital Design and Synthesis; Samir Palnitkar, Prentice Hall PTR; 2nd Edition

2. Fundamentals of Digital Logic with Verilog; Stephen Brown and Zvonko Vranesic; McGraw Hill, 2nd Edition

Reference Books

1. Digital Systems Design Using Verilog; Charles Roth, Lizy K. John, ByeongKil Lee; Cengage Learning 2nd Edition
2. A Verilog HDL Primer: J Bhaskar; Star Galaxy Publishing; 2nd Edition

List of Experiments

1. Design the following combinational circuits using Verilog HDL. Write a functional simulation test bench to test these designs. Generate all combinations of test inputs with a delay of 5 ns between the set of each test input. Observe the output in the console window.
   a. Full adder using operators
   b. Full adder using two Half adders
   c. 4:1 Mux using 2:1 Mux
   Synthesize the design and count the number of LUT. Plan a pin constraint for any one design for implementation on FPGA.

2. Compare the designs of a 16-bit ripple carry adder circuit and carry select adder. Write a post-implementation simulation test bench to test these designs. Generate random combinations of test inputs with a delay more than critical path delay between the set of each test input. Observe the output in the console window.
   Synthesize the design and count the number of LUT. Find the critical path delay. Plan a pin constraint for any one design for implementation on FPGA.

3. Model the 4-bit counter with the following specifications:
   a. Reset synchronous to clock positive edge.
   b. Counter has a 4-bit load terminal to count Up or down from this user-loaded value.
   Write a test bench to test this design and implement it on FPGA.

4. Design a 4-bit sequence detector using Verilog HDL. Write a test bench to test this design.

Create a Verilog RAM module with 8-bit wide data and 16 locations, initialize it with data, and perform a read operation to verify data retrieval. Find the average of the retrieved data in the console window

Course Code	25EE01TP0403-1			
Category	Programme Core Course			
Course Title	Data Structures and Algorithms using Python			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	IV

Course Outcomes

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

1. Understand the concepts of data structures.
2. Apply the concepts of linear (stacks, queues, linked lists) and non-linear (trees, graphs) data structures.
3. Implement different searching and sorting techniques.
4. Demonstrate the use and applicability of data conversion techniques.
5. Devise algorithms for solving real-world problems.

Syllabus

Module I: (5 Hrs)

Understanding data structures and algorithms, Python for data, Variables and expressions, Flow control and iteration, Overview of data types, objects and Python modules, Types of Data Structures – User defined, Built-in data types: List, Set, Dictionary, Tuple.

Module II: (5 Hrs)

Linear Data Structure – Arrays, Pointer structures, Nodes, Representation of arrays, Applications of arrays, Sparse matrix and its representation.

Module III: (6 Hrs)

Stack: Definitions & Concepts, Operations on Stacks, Applications of Stacks

Queue: Representation of Queue, Operations on Queue, Applications of Queue

Linked List: Singly Linked List, Doubly Linked List, Circular Linked List, Linked Implementation of Stack, Linked Implementation of Queue, Applications of Linked List.

Module IV: (7 Hrs)

Nonlinear Data Structure:

Tree – Definitions and Concepts, Representation of Binary Tree, Binary Tree Traversal (Inorder, Postorder, Preorder), Binary Search Trees

Graph – Representation of Graphs, Elementary Graph Operations (Breadth First Search, Depth First Search, Spanning Trees, Shortest Path, Minimal Spanning Tree).

Module V: (6 Hrs)

Sorting and Searching:

Insertion Sort, Quick Sort, Merge Sort, Heap Sort, Sorting on Several Keys, List and Table Sort, Linear Search, Binary Search.

Module VI: (6 Hrs)

Hashing and Symbol Tables:

Perfect Hashing Functions, Putting Elements, Getting Elements, Testing the Hash Table, Non-string Keys, Growing a Hash Table, Open Addressing.

Text Book:

- Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, *“Data Structures and Algorithms in Python”*, Wiley, 2013.

Reference Books:

1. Gowrishankar S, Veena A, *“Introduction to Python Programming”*, 1st Edition, CRC Press/Taylor & Francis, 2019. ISBN-13: 978-0-8153-9437-2.
2. Benjamin Baka, *“Python Data Structures and Algorithms”*, Packt Publishing Ltd., 2017.

Course Code	25EE01TP0403-2			
Category	Program Core Course			
Course Title	Data Structures and Algorithms using JAVA			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	III

Course Outcomes:

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

1. Understand the concepts of data structures.
2. Apply the concepts of linear (stacks, queues, linked lists) and non-linear (trees, graphs) data structures.
3. Implement different searching and sorting techniques.
4. Demonstrate the use and applicability of data conversion techniques
5. Devise algorithms for solving real-world problems.

Syllabus:

Module 1: (06 Hrs)

Data Structures and Algorithm Basics:

Introduction: basic terminologies, elementary data organizations, data structure operations; abstract data types (ADT) and their characteristics. Algorithms: definition, characteristics, analysis of an algorithm, asymptotic notations, time and space tradeoffs. Array ADT: definition, operations and representations – row-major and column-major.

Module 2: (06 Hrs)

Stacks and Queues:

Stack ADT: allowable operations, algorithms and their complexity analysis, applications of stacks – expression conversion and evaluation (algorithmic analysis), multiple stacks. Queue ADT: allowable operations, algorithms and their complexity analysis for simple queue and circular queue, introduction to double-ended queues and priority queues.

Module 3: (06 Hrs)

Linked Lists:

Singly Linked Lists: representation in memory, algorithms of several operations: traversing, searching, insertion, deletion, reversal, ordering, etc. Linked representation of stacks and queues, header node linked lists. Doubly and Circular Linked Lists: operations and algorithmic analysis.

Module 4: (06 Hrs)

Sorting, Searching and Hashing:

Sorting: Sorting: different approaches to sorting, properties of different sorting algorithms (insertion, Shell, quick, merge, heap, counting), performance analysis and comparison. Searching: necessity of a robust search mechanism, searching linear lists (linear search, binary search) and complexity analysis of search methods. Hashing: hash functions and hash tables, closed and open hashing, randomization methods (division method, mid-square method, folding), collision resolution techniques.

Module 5: (06 Hrs)
Trees:
Trees: basic tree terminologies, binary tree and operations, binary search tree [BST] and operations with time analysis of algorithms, threaded binary trees. Self-balancing Search Trees: tree rotations, AVL tree and operations, B+tree: definitions, characteristics, and operations (introductory).

Module 6: (06 Hrs)
Graphs:
Representation and Access: basic terminologies, representation of graphs, graph traversals: depth first search (DFS) and Breadth first search (BFS). Path Finding Algorithms: Dijkstra's Single Source Shortest Path (SSSP) algorithm, and Warshall-Floyd's All Sources Shortest Path (ASSP) algorithm. Spanning Trees: Introduction, minimum cost spanning trees, Prim’s Method and Kruskal’s Method for MSTs.

Text Book:

Data Structures and Algorithms in Java by Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, John Wiley & Sons, Inc., Sixth Edition, 2014

Reference Books:

1. Data Structures and Algorithms in Java, Robert Lafore, Second Edition, Sams Publishing, 2002

2. Thomas Cormen, Charles Leiserson, Ronald Rivest and Clifford Stein, Introduction to Algorithms, Third Edition; Prentice Hall of India; 2009

3. G. A. V. Pai; Data Structures and Algorithms: Concepts: Techniques and Application; First Edition; McGraw Hill; 2008.

List of Experiments:

1. Write Java programs that use both recursive and non-recursive functions for implementing the following searching methods: a) Linear search b) Binary search
2. Write Java programs to implement the following using arrays and linked lists
3. Write Java programs to implement the following using an array. a) Stack ADT b) Queue ADT
4. Write a Java program that reads an infix expression and converts the expression to postfix form. (Use stack ADT).
5. Write a Java program that uses both a stack and a queue to test whether the given string is a palindrome or not.
6. Write Java programs to implement the following using a singly linked list. a) Stack ADT b) Queue ADT
7. Write a Java program to perform the following operations: a) Construct a binary search tree of elements. b) Search for a key element in the above binary search tree. c) Delete an element from the above binary search tree.
8. Write a Java program to implement all the functions of a dictionary (ADT) using Hashing.
9. Write Java programs that use recursive and non-recursive functions to traverse the given binary tree in a) Preorder b) Inorder c) Postorder
10. Write Java programs for the implementation of bfs and dfs for a given graph.
11. Write Java programs for implementing the following sorting methods: a) Bubble sort b) Insertion sort c) Quick sort d) Merge sort e) Heap sort f) Radix sort g) Binary tree sort
12. Write a Java program to perform the following operations: a) Insertion into a B-tree b) Searching in a B-tree

Course Code	25EE01TH0405			
Category	Multidisciplinary Minor -2			
Course Title	Programming for Environmental IoT			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes

1. Explain the fundamental principles, classifications, and performance parameters of environmental sensors including sensitivity, specificity, accuracy, and linearity.
2. Differentiate between various types of actuators and describe their roles and complementary functions alongside sensors in environmental monitoring systems.
3. Demonstrate the ability to interface digital and analog sensors and actuators with Raspberry Pi, including push-buttons, displays, and cameras for environmental applications.
4. Develop environmental monitoring applications using ESP8266 by interfacing sensors, actuators, and peripherals, while implementing debouncing and edge detection.
5. Build and optimize IoT-based dashboards for remote environmental data visualization, while handling errors, reducing power and data usage, and ensuring reliable data transmission.

Module 1: Sensors for Environmental Monitoring

Classification of Sensors, Sensitivity, Specificity, Range, Precision, Accuracy and Resolution, Threshold and Linearity/non-Linearity

Module 2: Actuators for Environmental Monitoring

Definition and role of actuators in IoT, Types of actuators: electrical, mechanical, hydraulic, pneumatic, Actuators vs sensors: complementary functions

Module 3: Interfacing with Raspberry Pi

Understanding GPIO pins, Input/Output configuration, push-button interface, Debouncing and edge detection, Interfacing digital and analog sensors, actuators, display, camera for environmental applications

Module 4: Interfacing with ESP8266

Understanding GPIO pins, Input/Output configuration, push-button interface, Debouncing and edge detection, Interfacing digital and analog sensors, actuators, display, camera for environmental applications

Module 5: Displaying Sensor Data Remotely

Creating real-time dashboards on IoT platform, Triggering alerts or event, Error Handling and Optimization: Reconnecting on failure, Minimizing power and data usage, Data transmission intervals and timing.

Text Books:

1. Raspberry Pi for Python Programmers Cookbook, Tim Cox, Packt Publishing Limited; 2nd Revised edition, 2016.
2. Espressif Application notes ESP32/ESP8266/Node MCU

Reference Books:

1. Internet of Things Principles, Paradigms and Application of IoT, Joseph Kofi Wireko, Kaml Hiran, BPB Publications 2020 1st Edition
2. Raspberry Pi User Guide, EbenUptonand Gareth Halfacree, John Wiley& Sons, 2016.

13. Course Code	25EEOEC01TH0406			
Category	Open Elective			
Course Title	Open Elective-II/MOOCs/Industry Offered			
Scheme & Credits	L	P	Credits	Semester
	2	0	2	III

Note: MOOC Courses from Coursera platform

Course Code	25HS04PR0401			
Category	Co Circular Activity			
Course Title	Self Defence and Indian Marital Art			
Scheme & Credits	L	P	Credits	Semester
	0	2	0	IV

Course Type: Mandatory Audit Course (Practical)

Semester: All Ug Program during III and IV Semesters

Class Duration: 14 weeks (2 hours/week)

Mode of Instruction: In-person, practical, Practice and interactive sessions

Assessment: Satisfactory/Unsatisfactory based on participation, skill demonstration, and adherence to safety protocols.

Course Objective

This course aims to equip students—particularly female students—with the foundational skills and knowledge of essential self-defence techniques and Bhartiya Yudhkala (Indian Martial Arts). By the end of the program, students will possess increased awareness, confidence, and practical ability to protect themselves in potentially dangerous situations.

Course Description

"Self-Défense & Indian Martial Arts" is a life skills training program aimed at enhancing personal safety, especially for young women. It combines traditional Indian martial arts with modern self-defence techniques, focusing on physical fitness, mental resilience, and situational awareness.

Key Features:

- Real-life scenario-based training and escape techniques
- Use of everyday objects for defence
- Cultural insights into Indian martial arts
- Safe, structured learning environment

Course Outcomes

1. Demonstrate improved physical fitness, including enhanced strength, endurance, and agility.
2. Understand key self-defence laws and individual rights under Indian legal frameworks.
3. Apply basic defence techniques from Indian martial arts, including escape strategies and improvised weapon use.
4. Follow safety protocols and utilize resources such as protective gear and safe environments to manage emergencies and prevent injuries.

Weekly Schedule & Modules

Week	Module	Content
Week 1	Module 1 – Introduction	Overview of self-defence, importance, mindset training, and history of Indian martial arts.
Week 2	Module 1 – Introduction (contd.)	Basics of situational awareness and psychological preparedness.
Week 3	Module 2 – Targeting Weak Points	Anatomical knowledge for self-defence: eyes, nose, throat, groin, knees.
Week 4	Module 2 – Targeting Weak Points (contd.)	Practice and roleplay scenarios.
Week 5	Module 3 – Ground Défense	Techniques to defend and escape when on the ground.
Week 6	Module 3 – Ground Défense (contd.)	Partner practice and resistance drills.
Week 7	Module 4 – 20 Escape Techniques Through Indian Culture	Practical escape methods inspired by traditional Indian martial forms.
Week 8	Module 4 – 20 Escape Techniques (contd.)	Application in staged confrontation settings.
Week 9	Module 5 – Weapon of Opportunity	Training with improvised weapons: pen, keys, dupatta, bags, etc.
Week 10	Module 5 – Weapon of Opportunity (contd.)	Timing, precision, and legal considerations.
Week 11	Module 6 – Stick Rotation	Basics of stick (lathi/danda) handling and defence.
Week 12	Module 6 – Stick Rotation (contd.)	Practice with focus on flow, blocking, and disarming.

Note:

1. *Students of Higher Semester classes have to Complete this 30 hrs Audit Course during any one of the (III/IV) semesters. Preferably Similar distribution of Students as Sports Classes (Now Newly Named: Health-Fitness and Wellbeing) in First Year of UG program.*

2. *A minimum of 60% attendance and satisfactory performance is necessary for passing the courses.*

3. *The students who are Suffering from Serious Disease, injured, specially abled and medically Critical condition shall be exempted from the exam.*

4. *Requirement: A multipurposed Indoor Ac Hall with capacity of Minimum 200 Person for conducting Various, Self-Défense classes and Yoga & Meditation.* ***If there is enough space to conduct at least 03 Sections (200 students) of students simultaneously then it is feasible to start the course.***

5. Requirement of sufficient number of Expert Trainers of self-defence Course

Course Code	25EE01PR0408			
Course Type	Vocational and Skill Enhancement Course			
Course Title	Aptitude Development -1			
Scheme& Credits	L	P	Credits	Semester
		2	1	IV

Course Outcomes :
The student will be able to –
1. Improve analytical and logical reasoning ability.
2. Identify and evaluate deductive and inductive arguments.
3. Identify logical errors and false conclusions.
4. Improve aptitude, problem solving skills and reasoning ability.
5. Critically evaluate various real-life situations by resorting to analysis of key issues and factors.
6. Demonstrate various principles involved in solving mathematical problems and thereby reducing the
time taken for performing job functions.

Syllabus:

Section I
Coding Decoding, Direction Sense, Blood Relations, Analogy (word, letter, number, mixed), Ranking and
Ordering, Eligibility Testing, Syllogism, Inequalities, Sitting Arrangements, Clock and Calendar. Statements & Arguments, Statements & Course of Action, Cause and Effect, Cubes and Dice. Image Analysis (mirror & water images), Cubes and Cuboid, Error Detection, Grammar, Cloze Test, Comprehension, Double Fillers, Para jumbled sentences, One-word substitution.

Section II
Divisibility Rules, Numbers, Factors and multiples, Applications of HCF and LCM. Number Systems,
Linear Equations, Variation, Ages, Averages, Percentage, Ratio and Proportion. Simple Interest, Compound
Interest, Mensuration. Time & Work, Pipes and Cisterns, Boats and Streams, Partnerships, Problems on
Trains, working with different efficiencies, Work equivalence, Division of wages, Relative Speed, Problems based on Races. Percentages as Fractions and Decimals, Fundamental Counting principle, Basics of Permutation and Combination, Probability.

Text Books:
1. Shyam Saraf, Abhilasha Swarup, "Quantitative Apptitude and Reasoning", Cengage Publication
2. Dr. R. S. Aggarwal, "Quantitative Aptitude for Competitive Examinations", S. Chand Publications.
3. Dr. R. S. Aggarwal, "A Modern Approach to Logical Reasoning", S. Chand Publication

Reference Books:

1. Peeyush Bhardwaj, “TheHands-on Guide to Analytical Reasoning and Logical Reasoning”, Arihant Publication.
2. Arun Sharma, “How to Prepare for Logical Reasoning”, McGraw Hill Publication.
3. Nishit Sinha, “Logical Reasoning and DI”, Pearson Publication.
4. Moore, Parker, “Critical Thinking”, McGraw Hill Publication.
5. Arun Sharma, “How to Prepare for Quantitative Aptitude”, Tata McGraw Hill.
6. K. Sarvesh Verma, “Quantitative Aptitude Quantum Cat Common Admission Test”, Arihant Publications.

Course Code	25EE01HT0401			
Category	Honors Specialization			
Course Title	Single dimension signal processing on Edge			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes

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

1. Analyze and Interpret Signals.
2. Apply Digital Signal Processing Techniques.
3. Process Sensor Data on Edge Devices.
4. Develop Signal Processing Algorithms on Edge Platforms.
5. Solve Practical Problems Using Edge Devices

Module I: (5 Hrs)
Introduction to Signals Processing: Types of signals, Continuous-time and discrete-time domains, sampling theorem, quantization, analog to digital conversion, understanding discrete signals in time domain and frequency domain.

Module II: (5 Hrs)
Basics of Digital Signal Processing:
Representing discrete signals and systems in Z domain, basic operations on signals, algorithms for convolution, correlations, basic average and difference filter for single dimension signals

Module III: (5 Hrs)
Sensor Data Processing on Edge: Sensor data acquisition systems and protocols, collecting sensor data using Raspberry Pi, Feature extraction like mean, standard deviation, energy, and entropy, normalization, scaling, case studies for Edge applications – Driverless cars, Health monitoring, Smart City

Module IV: (5 Hrs)
Programming for Signal Processing: programming basic signal processing algorithms on edge platform: Raspberry pi, using signal processing libraries, implementing 1D convolution and cross-correlation, implementing average and difference filter on sensor data set for noise removal.

Module V: (10 Hrs)
Programming for Practical applications on Raspberry pi: Setup Raspberry pi for single dimension signal processing application, Case studies: Implementation of Heart Rate Monitoring using peak detection, Detect noise levels in an environment using signal's RMS value, Temperature Trend Analysis, Real-Time Respiratory Rate Monitoring, obstacle detection system, Motor Speed Control Based on Sensor Data

Text Book:

1. Digital Signal Processing, Nagoor Kani, 2nd Edition Mc-Graw Hill
2. Digital Signal Processing A Practitioner's Approach, Kaluri Venkata Rangarao, Ranjan K Mallik, John Wiley & Sons Ltd, 2005

Reference Books:

1. Digital Signal Processing: Principles, Algorithms, and Applications" by John G. Proakis and Dimitris G. Manolakis, 4e Pearson Education India

2. The Scientist and Engineer’s Guide to Digital Signal Processing, Steven W. Smith, California Technical Pub, First Edition
3. Make: Sensors: A Hands-On Primer for Monitoring the Real World with Arduino and Raspberry Pi , Tero Karvinen , Kimmo Karvinen , Ville Valtokari, 1st Edition , Make Community Publication.

Course Code	25EE10MT0401			
Course Type	Minor Specialization			
Course Title	Sensor Interfacing with Arduino and ESP8266			
Scheme& Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes:

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

1. Know about the open source hardware platforms like Arduio and ESP
2. Learn the working of sensors and actuators
3. Understand the working of serial communication protocols

Syllabus

Module I: (6Hrs)

Introduction to Arduino Prototyping Platform: Arduino IDE, Arduino C, Setting up the Arduino board, creating sketches,using Libraries, using example codes, Debugging Using the Serial Monitor.

Module II: (7Hrs)

Sensor interfacing with Arduino: Analog and digital sensors, Temperature sensors, Humidity sensors, Obstacle sensors, Ultrasonic sensor, Accelerometer and gyro, etc.

Module III: (7Hrs)

Serial Communication in Arduino: Serial and parallel communication, Serial communication protocols, UART, I2C, SPI, Wired and Wireless communication, Interfacing Communication Modules with Arduino.

Module IV: (7 Hrs)

Interfacing Displays and Actuators: 16x2 LCD, Graphical LCD, Graphical OLED, Arduino Tone functions, Melody generation on a piezo buzzer, Speed and direction control of DC, Servo and, Stepper Motor.

Module V: (6Hrs)

Introduction to ESP8266: ESP8266 development board, Programming ESP8266 through Arduino IDE, connecting to the internet, sending and receiving data on internet.

Module VI: (7Hrs)

Interfacing sensors and actuators with ESP8266: LDR, Temperature sensor, Humidity sensor, IR sensor, OLED, RGB LED, Servo motor, etc.

Text Books:

1. Arduino Cookbookby Michael Margolis, O’Reilly Media, Inc., 1st edition
2. Beginning C for Arduino By Jack Purdum (ebook)
3. Arduino for Beginners: Essential Skills Every Maker Needs, John Baichtal, PearsonEducation, Inc., 1st edition